MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE ABDELHAMID IBN BADIS - MOSTAGANEM

Faculté des Sciences Exactes et d'Informatique

Département de Mathématiques et informatique

Filière : Informatique

PROJET DE FIN D'ETUDES

Option : **Ingénierie des Systèmes d'Information**

THEME :

Les approches bio-inspirées appliquées à la classification des opinions

Etudiant(e) : « BERFAI ROUMISSA »

« BENSMAIN ZAHIRA »

Devant le jury :

Président : « SEHABA K. »

Examinateur : « MOUMEN M. »

Encadrant : « BESNASSI MILOUD »

Année Universitaire 2021-2022

Dédicace

Je dédie ce travail :

À mes très chers parents, pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études, leurs encouragements m'ont toujours donné de la force pour persévérer et pour prospérer dans la vie ;

Merci mama, merci papa.

À mon cher frère, Badredine pour son appui, son encouragement et ses conseils.

À mes âmes sœurs, mes jolies, Asma et Bochra pour leurs appui et soutien moral.

À mon petit Rayan.

À ma copine du cœur, Samira.

Et à tous ceux qui m'est cher.

Berfai Roumissa,

Dédicace

A mes parents bien-aimés, pour tous leurs sacrifices, amour, tendresse, appui et prières tout au long de mes études.

À mon frère et mes sœurs bien-aimées pour leur soutien,

Je remercie mon cher oncle Belkacem pour ses encouragements, sa gentillesse et sa bienveillance.

Je vous remercie d'être toujours là pour moi.

Bensmain Zahira,

Remerciement

On remercie Dieu le tout-puissant de nous avoir donné la santé et la volonté d'entamer et de terminer ce mémoire.

Tout d'abord, ce travail ne serait pas aussi riche et n'aurait pas pu avoir le jour sans l'aide et l'encadrement de Mr BESNASSI MILOUD, on le remercie pour la qualité de son encadrement exceptionnel, pour sa patience, sa rigueur et sa disponibilité durant notre préparation de ce mémoire.

Nous tenons à exprimer notre profonde gratitude et nos sincères remerciements aux :

Membres de jury qui nous ont fait l'honneur de réviser ce travail ;

Toutes les personnes qui nous ont aidés, de près ou de loin.

Notre remerciement s'adresse également à tous nos professeurs pour leurs générosités et la grande patience dont ils ont sues faire preuve tout au long notre formation.

Résumé

Avec l'avènement des réseaux sociaux sur le WEB, L'analyse de sentiments est apparue comme l'un des nouveaux défis en traitement automatique des langues. Il s'agit d'une tâche difficile et très importante qui implique le traitement des langues naturelles et l'apprentissage machine.

Par rapport à l'analyse de sentiments dans la plupart des langues latines, la langue arabe est plus difficile à analyser en raison de sa complexité morphologique, ses particularités et la grande variation de ses dialectes. Il est donc nécessaire de proposer des solutions et de développer des programmes de catégorisation à l'aide de différentes techniques de classification.

Dans notre projet nous avons adapté une approche qui se base sur la méthode de fouille de données à savoir la classification ainsi que ses techniques qui s'inspirent de la biologie en l'occurrence les algorithmes bio inspirés, sur un ensemble de données public: AJGT (Arabic Jordanian Géneral Tweet). Ce travail nécessite deux phases. La première est la phase de prétraitement, qui comprend les étapes de Nettoyage des données, Normalisation, Tokenization, Suppression des mots vides, stemming. La seconde est l'application des trois classificateurs : Support Vector Machines (SVM), Random Forest (RF), K Nearest Neighbors (KNN).Et la sélection d'attributs à l'aide de trois techniques différentes: inspirée de la génétique (Algorithm genetic AG), basée sur l'imitation du comportement des faucons (Harris Hawks Optimizer HHO), et basée sur l'imitation du comportement des loups gris (Grey Wolf Optimizer GWO).

D'après notre étude, l'AG à base du classifieur k-NN a montré une meilleure performance pour la classification des opinions en termes d'accuracy, précision, Taux de rappel et F-score.

Mots-clés

Analyse des sentiments, Opinion mining, Apprentissage profound, Classification du Texte, Catégorisation du Texte, Traitement Automatique de la Langue Naturelle (TALN), Approches bio-inspirées, Sélection d'attributs, Genetic algorithm (GA), Grey wolf optimization (GWO), Harris Hawks optimization (HHO)

Abstract

With the advent of social networks on the WEB, sentiment analysis has emerged as one of the new challenges in natural language processing. This is a difficult and very important task that involves natural language processing and machine learning.

Compared to the analysis of feelings in most Latin languages, the Arabic language is more difficult to analyze because of its morphological complexity, its peculiarities and the great variation of its dialects. It is therefore necessary to propose solutions and develop categorization programs using different classification techniques.

In our project we have adapted an approach based on the data mining method, namely classification as well as its techniques which are inspired by biology, in this case bio-inspired algorithms, on a public data set: AJGT (Arabic Jordanian General Tweet).

This work requires two phases. The first is the pre-processing phase, which includes the steps of Data Cleansing, Normalization, Tokenization, Stopword Removal and stemming. The second is the application of the three classifiers: Support Vector Machines (SVM), Random Forest (RF), K Nearest Neighbors (KNN).And the selection of attributes using three different techniques: inspired by genetics (Algorithm genetic AG), based on the imitation of hawk behavior (Harris Hawks Optimizer HHO), and based on the imitation of behavior of gray wolves (Grey Wolf Optimizer GWO).

According to our study, the kNN classifier-based GA showed better performance for opinion classification in terms of accuracy, precision, recall and F-score.

Keywords:

Analyse des sentiments, Opinion mining, Apprentissage profound, Classification du Texte, Catégorisation du Texte, Traitement Automatique de la Langue Naturelle (TALN), Approches bio-inspirées, Sélection d'attributs, Genetic algorithm (GA), Grey wolf optimization (GWO), Harris Hawks optimization (HHO)

Liste des figures

Liste des tableaux

Liste des abréviations :

Abréviation	Expression complète	page
SVM	Support Vector Machines	03
SA	Sentimen Analysis	03
NB	Naive Bayes	03
TFIDF	Term Frequency-Inverse Document Frequency	03
ACP	Analyse en composantes principales	04
GI	Gain ınformation.	04
ASM	Arabe Standard Moderne.	04
BM	Modèle binaire.	04
Tf	Term Frequency	04
SGD	Stochastic Gradient Descentla descente	05
MNB	Mutlinomial Naive Bayes	05
DT	Decision tree	05
LABR	Large Scale Arabic Book Reviews	05
AJGT	Arabic Jordanian General Tweets	05
SSA	Salp Swarm Algorithm	06
GWO	Grey Wolf Optimizer	06
DE	Differential Evolution	06
PSO	Particle Swarm Optimizer	06
GA	Genetic Algorithm	06
WOA	Whale Optimization Algorithm	06
TALN	Traitement automatique du langage naturel.	07
KNN	K Nearest Neighbors	08
ISRI	The information science research instit	24
RF	Random Forest	28
HHO	Harris Hawks Optimisation	41
TP	True Positive	57
TN	True Negative	57
FP	False Positive	57
FN	False Negative	57

Table des matières

Introduction générale :

Les gens s'appuient généralement sur les expériences passées et les connaissances des autres, c'est pourquoi ils demandent souvent des recommandations avant de prendre une décision dans n'importe quel domaine.

Avec la disponibilité croissante des informations et ressources d'opinions tel que les réseaux sociaux (par exemple Twitter, Facebook, des blogs, des micro-blogs, des commentaires, des statuts et publications dans des sites de réseaux sociaux), le capture d'opinion publique sur des stratégies d'une entreprise, des événements sociaux, des campagnes de marketing, des mouvements politiques ou d'un gouvernement, des produits et des services recueille devient une fortune et un intérêt croissant de la communauté scientifique ainsi que du monde des affaires. Mais l'accès aux ces données d'une façon précise et rapide devient très difficile, et l'analyse manuelle est presque impossible, même s'il est possible elle infect l'efficacité, la rapidité et le coût. Ce qui nécessite de développer des programmes de catégorisation automatique.

Dans notre étude nous appuierons sur la classification des opinions arabe exprimées à travers les textes. Il comporte quatre chapitres :

- Dans le premier chapitre, un état de l'art de l'analyse des sentiments et ses trois approches commencerons par les approches basées sur les techniques d'apprentissage machine, puis les approches basées sur le lexique, finissons par les approches basées sur les méthodes hybrides qui combinent les deux approches précédentes.
- Dans le deuxième chapitre, nous allons présenter les particularités de la langue arabe, sa morphologie et les différentes difficultés de traitement automatique de cette langue. Puis nous allons expliquer les prétraitements nécessaires à appliquer sur le texte, l'apprentissage automatique et en fin une présentation des algorithmes de classification.
- Dans le troisième chapitre, nous allons introduire quelques notions dédiées aux algorithmes inspirés du nature tell que les algorithmes évolutionnaires à base de biologie
- Dans le quatrième chapitre, une description de corpus utilisé pour réaliser notre système sera représentée, ainsi les résultats expérimentaux.

1 Chapitre 1 Etat de l'art

1.1 Introduction

L'utilisation incontournable de l'internet dans toutes les activités quotidiennes ou les domaines professionnels et la croissance rapide des réseaux sociaux dans la société arabe, engendre une augmentation énorme du volume d'informations et de documents arabe numériques sur le web.

Cette augmentation coïncide avec l'importance du domaine de l'analyse des sentiments qui a gagné encore plus de valeur et devient l'une des nouveautés les plus intéressantes de ces dernières années. C'est pour cela en trouve plusieurs travaux qui sont réalisés en tous les domaines d'application connus et avec différents sous-objectifs.

Nous nous intéressons dans ce chapitre aux travaux relatifs à l'analyse des sentiments arabes, nous présentons un état de l'art de la fouille d'opinions ou l'analyse des sentiments en langue arabe, et de ses différentes approches. La première concerne les méthodes d'apprentissage automatique qui consistent à classer les documents à partir d'une base de données d'apprentissage. La deuxième concerne les méthodes de classification basée sur le lexique qui repose sur un lexique des sentiments, une collection de termes de sentiments connus et précompilés. Et la troisième qui combine les deux approches précédentes.

1.2 Analyse des sentiments

La fouille d'opinions est l'un des domaines de recherche les plus actifs en traitement du langage naturel depuis le début des années 2000 [1]. Le but de l'analyse des sentiments est de définir des outils automatisés capables d'extraire des informations subjectives de textes en langage naturel tels que : les opinions et les sentiments pour la création de connaissances structurées pouvant être utilisées par les systèmes d'aide à la décision ou les décideurs. Selon le domaine d'application, plusieurs noms sont utilisés pour l'analyse des sentiments comme: opinion mining [2], sentiment analysis [3]. Celles-ci s'accompagnent souvent de problèmes de classification des textes d'évaluation tels que ceux disponibles sur Amazon ou Epinions. Afin de décider de l'orientation d'un document [4], en fonction de leur polarité : Positif / Négatif /

Neutre par rapport au sujet d'intérêt [5], [6]. Dans cette section, nous décrivons une synthèse des travaux relatifs à l'analyse des opinions pour les données issues des réseaux sociaux.

Dans la littérature, trois approches ont été adoptées en se basant sur l'apprentissage machine, l'analyse lexicale et les techniques hybrides en combinant les deux.

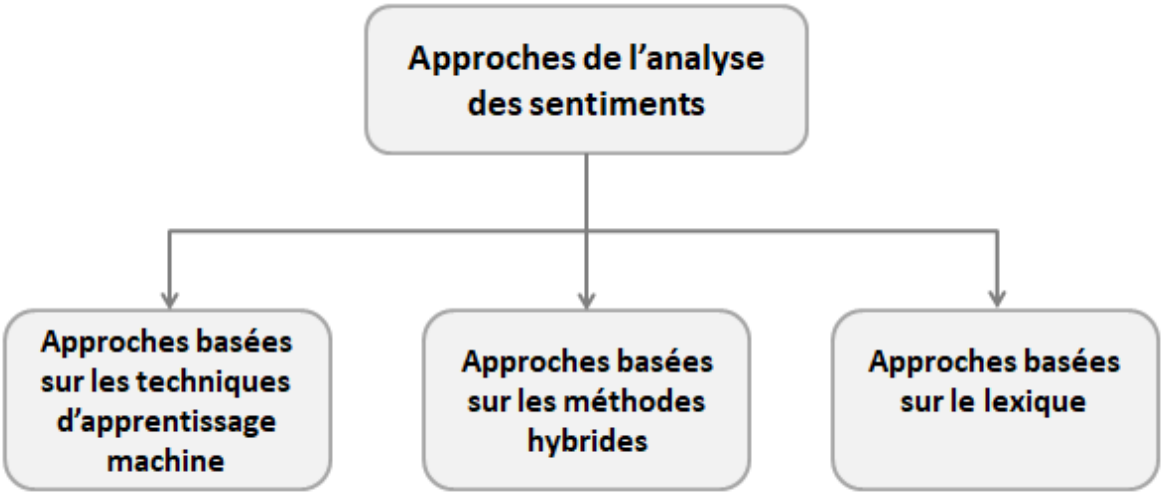

Figure 1-1 – Approches de l'analyse des sentiments

1.2.1 Les approches basées sur les techniques d'apprentissage machine

Dans le cadre d'une recherche menée par Al-Kabi et al en 2013 [7], ils ont créé un nouvel ensemble de données arabe collecté auprès de YahooMaktoob, uniquement pour le sentiment analysis (SA). Et ils ont testé deux classificateurs d'apprentissage machine contenant Support Vector Machines (SVM) et le Naive Bayes (NB) sur cet ensemble de données. Dans le prétraitement, la technique de pondération de la TFIDF (Term Frequency-Inverse Document Frequency) a été utilisée, suivie par la phase de la sélection des caractéristiques afin de réduire le nombre de caractéristiques. Ils ont mené leurs expériences en combinant chaque classificateur avec des caractéristiques sélectionnées et pondérées. En conséquence, le taux de reconnaissance déclarée du classificateur SVM atteint 68,2 %, ce qui est supérieur à l'accuracy du classificateur NB.

Dans le travail d'Abdulla et al [8]**,** ils ont créé un nouvel ensemble de données collectées auprès de Yahoo-Maktoob sur plusieurs sujets arabes tels que la science, l'art, le social, la politique et la technologie. Ils ont comparé la performance entre les classificateurs du SVM et NB, on appliquant la pondération TFIDF comme technique d'extraction des mots

clés pour chaque classificateur. La reconnaissance rapportée de l'application du classificateur SVM a surpassé le classificateur NB avec un taux d'accuracy de 64,1 %.

Dans [9], les auteurs ont utilisé trois classificateurs d'apprentissage machine, K-NN (k-Nearest Neighbours), SVM et NB en utilisant l'ensemble de données de l'OSA. Ils ont combiné chacun des classificateurs utilisés avec une technique de sélection des caractéristiques à partir des différentes techniques de sélection de caractéristiques telles que : ACP (Analyse en composantes principales), GI (gain information). Ils ont également appliqué le stemming sur les éléments sélectionnés et ils ont supprimé les stops words. Sur la base de leurs expériences, ils ont conclu que l'utilisation des SVM dans la sélection d'attributs atteint un taux d'accuracy de 92,4 %.

L'un des nombreux traveaux qui ont été réalisés pour analyser et classer les sentiments de l'arabe et ses dialectes en utilisant des approches supervisées est le travail de Cherif et al en 2015 [10], qui a fait appel à la technique SVM pour classer les sentiments de message écrit en arabe (ASM : l'Arabe Standard Moderne) en cinq classes. Pour réaliser cette tâche, les auteurs ont commencé par le prétraitement des phrases. Ils font également appel à un extracteur de lemmes présenté précédemment par [10] afin de supprimer les préfixes et suffixes des mots pour obtenir leurs radicaux. Cependant, Il faut noter que les auteurs suppriment les préfixes et les suffixes reliés à la conjugaison, au pluriel et aux pronoms et ils gardent les affixes reliés à la négation qui pourraient affecter la qualité de l'analyse de sentiments.

Dans le cadre d'une recherche menée par Duwairi et al en 2016 [11], ils ont étudié l'utilisation de trois différents classifieurs d'apprentissage automatique : le SVM, le K-NN et le NB, à partir de leurs propres donnés sur les sentiments arabes. L'ensemble de données utilisé a été recueilli sur Twitter, qui contient 2591 tweets en arabe. En outre, ils ont étudié différentes combinaisons entre les trois classificateurs et les trois différents types de techniques de pondération.

Les techniques de pondération utilisées sont la fréquence des termes (TF), Modèle binaire (BM) et TFIDF. Ils ont indiqué que le NB avec la technique de pondération TF a surpassé les autres avec un taux accuracy de 69,97 %, mais la combinaison du classificateur SVM et de la technique de pondération TFIDF a surpassé les autres classificateurs en termes de précision. En outre, ils ont étudié différentes combinaisons entre les trois classificateurs et les trois différents types de techniques de pondération.

Plusieurs techniques ont été utilisées pour la SA (Sentiments analysis) arabe. Par exemple, dans [12], ils ont employé cinq classifieurs d'apprentissage machine contenant le SVM, SGD (Stochastic Gradient Descentla descente), MNB (Mutlinomial Naive Bayes), les arbres de decision DT (decision tree), et NB (Naive Bayes) sur un grand ensemble de données de LABR (Large Scale Arabic Book Reviews).

Selon les résultats obtenus, le classificateur MNB a mieux performé que les autres classificateurs utilisés. Ils ont employé différents modèles d'extraction de caractéristiques en se basant sur les classifieurs cités précédemment, et l'étude expérimentale a montré que la meilleure performance est obtenue par le classifieur MNB en utilisant l'unigramme.

Enfin, ils ont introduit les algorithmes génétiques comme nouvelle contribution pour sélectionner les caractéristiques pertinentes au classifieur MNB, ce qui a permis d'améliorer ses performances avec un taux d'accuracy de 85 %.

Dans [13], les auteurs ont utilisé deux méthodes SVM et BN (Naïve Bays) pour classifier un ensemble de messages en positif, négatif ou neutre. Pour ce faire, ils ont construit un corpus arabe ASM (l'Arabe Standard Moderne) contenant 3700 messages extraits de Twitter. Chaque message a été annoté par trois locuteurs arabes natifs en positif, négatif et neutre.

Dans le cadre d'une recherche menée en 2017 par Alomari, et al [14], ils ont créé un nouvel ensemble de données SA en arabe appelé Arabic Jordanian General Tweets (AJGT). Ils ont comparé les performances des classificateurs SVM et NB en utilisant différentes combinaisons de prétraitement. Le nouvel ensemble de données a été collecté de Twitter sur différents sujets jordaniens avec un total de 1800 tweets. Ils ont également étudié et comparé les performances des deux classificateurs en utilisant plusieurs techniques de prétraitement. Plus exactement, ils ont comparé trois techniques d'extraction de caractéristiques à base de Ngrammes comme les Bigrams, les Unigrams et Trigrams. Par ailleurs, ils ont combiné les techniques d'extraction de caractéristiques avec la technique de pondération TF/ TFIDF. Ils ont signalé que le classificateur SVM combiné avec la technique de pondération TFIDF a surpassé la combinaison des autres scénarios et elle a atteint un taux d'accuracy de 88,72% en termes de F-mesure.

Dans [15], les auteurs ont mené une recherche impliquant l'utilisation d'un ensemble de classificateurs d'apprentissage automatique basée sur l'algorithme de vote majoritaire en conjonction avec quatre classificateurs, comprenant Naive Bayes, Support Vector Machines,

Decision Trees et K-Nearest Neighbor, pour analyser et classer les polarités des sentiments arabes. Les expériences ont montré que l'ensemble des classificateurs d'apprentissage automatique donne de meilleurs résultats en comparant avec les classificateurs de base. Ils ont constaté que la méthode de vote représente l'approche de classification la plus efficace pour l'analyse des sentiments du texte arabe. Elle utilise différents classificateurs pour classer chaque cas. Le vote à la majorité des décisions de tous les classificateurs est combiné pour prédire la classe de l'instance testée.

Dans [16], ils ont proposé une version améliorée de l'Algorithme d'optimisation des baleines (WOA), appelé (IWOA), pour le problème de la sélection des caractéristiques dans l'analyse du sentiment arabe. Les expériences ont montré que l'IWOA présente nettement une meilleure performance par rapport aux autres algorithmes d'optimisation comme GWO (Grey Wolf OptimizerGrey Wolf Optimizer), DE (Differential Evolution), PSO (Particle Swarm Optimizer) et GA(Genetic Algorithm) en termes de reconnaissance et de nombre de caractéristiques sélectionnées. Dans leur approche, ils ont fusionné deux phases de réduction dimensionnelle. La première phase a utilisé un filtre basé sur la métrique GI puis ils ont appliqué l'IWOA. Ils ont testé IWOA sur le corpus AJGT, le taux de classification obtenu était de 87.61% avec un nombre d'attributs sélectionnés qui vaut 1102.

Dans [17] des algorithmes inspirés de la théorie de swarm intelligence ont été développés pour pouvoir appliquer la sélection d'attributs. Dans ce cadre, nous trouvons l'algorithme SSA (Salp Swarm Algorithm) qui a joué un grand rôle dans l'amélioration de l'analyse des opinions de la langue arabe [18].

En premier lieu, les auteurs ont appliqué le filtrage à base de GI ensuite Ils ont testé la méthode SSA sur le corpus AJGT et ils ont obtenu un taux de reconnaissance de 80.08% et en gardant un nombre de termes de 832. Dans le même contexte**,** le travail de Boudjnane et Kadri [19], ils ont traité la classification de texte arabe à l'aide de la méthode d'optimisation GWO (Gray Wolf Optimizer) sur plusieurs corpus en arabe tels que (AJGT, OSAC, BBC, CNN etc…), et ils ont quantifié leurs résultats de classification en termes de l'accuracy et le taux de sélection d'attributs. L'étude expérimentale a montré une performance meilleure pour les corpus qui contiennent des classes bien équilibrées.

1.2.2 Les approches basées sur le lexique

Les méthodes d'apprentissage machine ont été largement utilisées lors de l'analyse des opinions dans les réseaux sociaux. Cependant, l'analyse de sentiments basée sur le lexique

semble attirer plus d'attention. Montejo-Ráez et al [20] Ont proposé une nouvelle approche pour détecter la polarité des messages twitter ; basée sur la combinaison de la pondération des synsets du texte par un algorithme de promenade aléatoire (random walk en anglais) et les scores de polarité fournis par entiWordNet. Les tests ont montré la possibilité de construire un système comparable à une approche supervisée basée sur l'algorithme Machine à vecteurs de support en termes de performance (62,85% F-score). Cependant, les chercheurs n'ont pas considéré des éléments qui pourraient biaiser l'analyse tels que la négation et l'élimination des mots vides.

Le travail [21] a permis de construire un lexique de 120 000 termes arabes ASM (l'Arabe Standard Moderne). Pour aboutir à ce dernier, les auteurs ont commencé par collecter des lemmes en arabe. Ils les ont traduits en anglais en utilisant Google traduction. Puis la suppression des mots répétés. Ces auteurs n'ont pas pris en considération le contexte du lemme dans le processus de traduction. Ils utilisent ensuite un lexique de sentiments anglais pour déterminer leur valence et intensité.

Chiavetta et al [22] Ont présenté un système qui classifie automatiquement l'orientation de sentiments exprimée dans les critiques de livres. Le système est basé sur une approche lexicale et utilise les techniques de traitement automatique du langage naturel (TALN) pour prendre en compte la relation linguistique entre les termes. La classification de la polarité globale du texte est basée sur la force de sentiment moyenne de ses phrases, tandis que la classification de chaque phrase est obtenue par un processus d'analyse inspectant, pour chaque terme, une fenêtre d'items précédents pour détecter des combinaisons particulières d'éléments donnant des inversions ou variations de polarité. Le système proposé est capable de classer automatiquement les révisions positives et négatives, avec une précision moyenne de 82%.

Mertiya and Singh [23] Ont utilisé des critiques de films comme ensemble de données pour la formation et les tests. Ils ont levé l'ambiguïté de classification de l'algorithme Naïve Bayes en intégrant l'analyse des adjectifs. Premièrement, ce modèle est appliqué sur les tweets collectés, ce qui donne des ensembles de tweets correctement et faussement étiquetés. Le faux ensemble étiqueté est ensuite traité avec une analyse d'adjectif pour déterminer sa polarité.

Les résultats expérimentaux montrent que la précision globale du processus est améliorée par rapport à l'algorithme classique de Naïve Bayes.

1.2.3 Les approches basées sur les méthodes hybrides

Afin de réunir les avantages des approches basées sur le lexique et l'apprentissage automatique, des chercheurs ont proposé des modèles hybrides combinant les deux approches.

Le travail rapporté par El-Halees [24] a évalué la reconnaissance de l'analyse du sentiment arabe lors de l'utilisation des approches basées sur le lexique et d'apprentissage automatique. Pour la première approche, il a construit un dictionnaire manuellement de mots subjectifs arabes, et dans l'apprentissage automatique, il a utilisé l'entropie maximale, les k-plus proches voisins, Naïve Bayes et les algorithmes de machine à vecteurs de support. Ces classificateurs ont atteint une précision maximale de 63% avec une marge d'erreur de 17%, la meilleure accuracy obtenue par K-NN atteint 63,58%.

Dans [25], les auteurs ont comparé deux approches pour l'analyse des sentiments arabe incluant une approche basée sur un dictionnaire et une autre basée sur le lexique (non supervisée). Tout d'abord, ils ont créé leur propre ensemble de données en composant 2000 tweets en arabe sur divers sujet. En outre, ils ont construit leur propre lexique arabe en téléchargeant 300 Mots anglais du site web de SentiStrength et ils ont traduit les mots anglais collectés en mots arabes correspondants.

Ils ont amélioré le lexique en ajoutant les synonymes de ces 300 mots traduits. Ils ont également étudié la comparaison entre quatre classificateurs d'apprentissage machine comme SVM, KNN NB, et arbre de décision pour l'analyse des sentiments arabe en considérant les trois types de «stemming » telles que : light stemming, root stemming, ou no stemming. D'autre part, ils ont également mené l'expérience en utilisant le lexique construit sur l'ensemble des données recueillies sur Twitter en arabe. Ils ont signalé que la reconnaissance de l'approche fondée sur un dictionnaire a surpassé l'approche fondée sur le lexique. En outre, la meilleure précision de l'approche fondée sur un dictionnaire et le classificateur SVM était avec un taux d'accuracy de 87,2%, tandis que la reconnaissance atteinte par l'approche fondée sur le lexique était de 59.6%.

Dans [26], les auteurs ont proposé une méthode hybride fondée sur un lexique et un classificateur NB en même temps. La méthode proposée est précédée d'une phase de prétraitement (normalisation, segmentation, etc.). Le lexique intervient pour remplacer les mots avec leurs synonymes. Ces auteurs se focalisent sur l'arabe standard moderne (ASM).

Gamallo and Garcia [27] ont proposé une famille de classificateurs basée sur Naïve Bayes pour détecter la polarité des tweets. Les expériences ont montré que la meilleure performance est obtenue en utilisant un classificateur binaire conçu pour détecter seulement deux catégories : positives et négatives. Afin de détecter les tweets avec et sans polarité, le système utilise une règle très basique qui recherche les mots de sentiment dans les tweets à analyser. Les auteurs ont construit un lexique de polarité avec des entrées positives et négatives provenant de différentes sources telles que : AFINN-11, Hedonometer et Sentiwordnet 3.0. Le modèle hybride obtient un F-score de 63%.

1.3 Conclusion

Ce chapitre a introduit une étude de la littérature sur la classification des sentiments. L'approche utilisée est basée sur l'apprentissage automatique, les dictionnaires ou une combinaison des deux. La plupart de ces techniques présentent encore quelques inconvénients lorsqu'il s'agit de méthodes d'apprentissage automatique qui utilisent des données d'entraînement provenant de domaines spécifiques ou de dictionnaires généraux qui ne tiennent pas compte du contexte étudié.

Nous avons également noté, qu'il existe quelques travaux de recherche qui utilisent à la fois les approches bio-inspirées avec la classification des opinions de la langue arabe.

2 Chapitre 2 Prétraitement et Classification

2.1 Introduction

Plusieurs travaux qui consistent au traitement automatique de la langue arabe ayant des difficultés à appliquer les différentes tâches pour cette analyse telle que les prétraitements et la classification à cause aux complexités de ses caractéristiques lexique et morphologique.

Dans ce chapitre nous commencerons par présenter les caractéristiques de la langue arabe, ses particularités, sa morphologie et les différentes difficultés de traitement automatique de cette langue. Puis nous expliquerons les prétraitements nécessaires à appliquer sur le texte, l'apprentissage automatique et comment les appliqués dans le traitement automatique, en finissant par une présentation des algorithmes de classification.

2.2 La particularité de la langue arabe

La langue arabe est une langue morphologiquement complexe qui possède des caractéristiques bien particulières par rapport aux autres langues dont :

- **Alphabet arabe**

Contrairement aux langues latines, La langue arabe est une langue sémitique qui s'écrit de droite à gauche et dont l'alphabet est un abjad [28]. Et les notions de lettre capitale et lettre minuscule n'existent pas. Son alphabet comprend essentiellement par des consonnes et ainsi des voyelles « ا » « ي » Et « و » Les deux dernières sont des réalisations contextuelles des « و » et « ي »glides.

L'écriture arabe comporte également des voyelles qui ne sont pas essentielles à l'écriture ainsi qu'un certain nombre de signes annexes dont l'emploi est facultatif hormis pour le coran servant à noter les trois voyelles brèves ("ً " (a), " ٌ" (u) et " ٍ" (i)).

Il existe de plus une série d'autres diacritiques de syllabation dont les plus courantes sont l'indication de l'absence de voyelle " ْ " (sukūn) et la gémination des consonnes " ّ " (šhadda).

Notons aussi que si un mot arabe est indéfini (sans article ni complément du nom), il prend (sauf exceptions) les désinences " ً " (an), " ٌ " (un) où " ٍ" (in), nommées nounations ou

tanwīn, Celles-ci sont notées par des diacritiques spéciales marquées par le redoublement du signe de la voyelle qui précède le suffixe « n » attendu en fin de mot.

L'écriture arabe est dite monocamérale. En addition, l'arabe est une langue semi-cursive, la plupart des lettres s'attachent entre elles, leurs graphies diffèrent selon qu'elles soient précédées et/ou suivies d'autres lettres ou qu'elles soient isolées. Seulement six d'entre-elles ne s'attachent jamais à la lettre suivante, ils sont : «"ز" "ر" "ذ" "د" "و" "ا" ».

- **Voyelle**

Trois types de voyelles existent en langue arabe, les voyelles longues et les voyelles courtes.

- **Les voyelles longues**

Les trois lettres (ا، و، ي) qui s'insèrent dans le mot exactement comme les consonnes.

- Le « Alif » (ا → /a/) Ex : «حال » (état) ;
- Le « Ya » (ي → /i/) Ex : «فيل» (éléphant) ;
- Le « Waw » (و → /ou/) Ex : «فول» (fèves).

- **Les voyelles brèves**

Les voyelles brèves (diacritiques) : l'écriture arabe utilise des signes diacritiques marqués comme des voyelles courtes. Ceux-ci sont prononcés brièvement et placés 'au-dessus' ou 'au-dessous' des lettres pour fournir la prononciation correcte et clarifier le Sens du mot [Tableau 2-1].

Tableau 2-1 – Les diacritique

Voyelle brève	Nom	Description	Situation
َ	فتحة/fathatun/	A	Au-dessus
ُ	ضمة/ Dhamatun/	Ou	Au-dessus
ِ	كسرة/kasratun/	I	Au-dessous
ْ	سكون/sokûnun/		Au-dessus

Mais la majorité des textes arabe sont écrits sans signes diacritiques, Cependant, la majorité des textes présente un problème d'ambiguïté lexicale qui remet en cause le calcul systèmes, comme l'exemple du [Tableau 2-2].

Tableau 2-2 – Interprétation du mot مدرسة

Unité lexicale	1ère interprétation		2ème interprétation		3ère interprétation	
مدرسة	مَدْرَسَةٌ	École	مُدَرِّسَةٌ	Enseignante	مُدَرَّسَةٌ	Enseignée

La plupart des études dans le traitement automatique de la langue arabe ignorent les signes diacritiques et les suppriment durant une phase préalable qu'on appelle la normalisation qui consiste aussi à remplacer quelques lettres par d'autres selon des règles prédéfinies.

- **Doubles voyelles**

Ce type permet de redoubler la consonne ou bien de créer une tonalité à la fin du mot associé selon le cas :

- **La Shadda** : C'est le signe "ّ " (šhadda) de la gémination en arabe.il représente un doublement d'une consonne lors de sa prononciation et ne peut être utilisé dans la 1ère lettre d'un mot Exemple : «دَرَّسَ» a enseigné « دَرَسَ»a étudié »
- **Le tanwin :** C'est des signes qui sont utilisés à la fin des mots indéterminés consistant à un doublement des signes diacritiques et qui produisent le même son que les trois premières voyelles simples avec l'ajout du son « n » à la fin.
- **La Hamza (ء)** : écrivez en quatre types selon des règles spécifiques.

Sur « ا (Alif) » أ Ex : سَأَل | فَأْس

Sur «و (Waw) » ؤ Ex : تَفَاؤُل | مُؤَنث | بُؤْس

Sur «ي (Ya) » ئ Ex : بيئَة | شاطِئ

Sur la ligne ء Ex : سماءٌ | هواءْ

2.3 Morphologie du mot arabe

La morphologie est l'étude des structures des mots (unités lexicales) ainsi que leurs formes. Elle concerne l'étude de la structure morphologique des mots considérés isolément (hors contexte) sous le double aspect de la nature et des variations qu'ils peuvent subir [29]. L'analyse morphologique est substantielle pour chaque système de traitement automatique de la langue naturelle. Elle a pour objectif de regrouper les mots en classes utilisables par les

autres niveaux d'analyse de telle façon que chaque classe peut être associée par une étiquette appelée catégorie grammaticale ou catégorie lexicale [30].

Bien que des différents travaux de recherche appuyaient sur l'analyse morphologique des langues latines. Mais l'application est considérée difficile pour des langues qui ont une morphologie riche et complexe telle que la morphologie arabe ; même si nous avons raccourci le lexique aux seules informations non calculables (i.e. forme canoniques, racines, etc.) que nous utilisons des règles pour connaître le reste des informations [31].

2.3.1 Les éléments essentiels de la Morphologie Arabe

- **Les racines**

La racine est la plus petite unité lexicale qui permet de former un mot ; Le lexique arabe comprend trois catégories de mots : verbes, noms et particules.

Pour les verbes à l'aide de différents schèmes une famille de mots peut ainsi être générée d'un même concept sémantique à partir d'une seule racine. Par ce phénomène caractéristique nous disons donc que l'arabe est une langue à racines réelles à partir desquelles, nous déduisons le lexique arabe.

- **Les affixes**

Les affixes sont des morphèmes qui s'ajoutent au début (les préfixes) ou à la fin des mots arabes (les suffixes), Ils permettent de former, à partir d'une même racine, de nouveaux lemmes.

En général, Ils sont utilisés pour accorder aux mots des éléments syntaxiques. Ils marquent l'aspect verbal, le mode, les propriétés transitives.

- **Le schème**

Le schème joue un rôle très important dans le processus de génération des formes dérivées à partir d'une racine ou d'extraire cette dernière à partir d'un mot, il représente une forme ou modèle général composé de trois consonnes ف[f], ع [’] et ل[l] qui sont vocalisées et qui peuvent être augmentées par d'autres lettres (préfixe, suffixe et infixe).

- **Les stems**

Un Stem est la dérivation obtenue à partir d'une racine donnée selon un modèle (un schème). L'arabe classique à un grand nombre de Stems qui ne sont pas tous utilisables [32]**.**

Le Stem correspond à un modèle si et seulement s'il possède le même nombre de lettres et les mêmes lettres dans les mêmes positions.

- **Les mots dérivés**

La plupart des mots arabes sont considérés comme des mots dérivés, puisqu'ils sont construits à partir des racines.

2.3.2 La catégorie d'un mot

Dans la langue arabe nous avons trois catégories de mot : le nom, le verbe et les particules.

- **nom**

Est une entité ou un élément qui exprime un sens indépendamment du temps pour Désigner un objet ou un être. Les substantifs arabes sont de trois catégories, Les primitifs qui représentent les noms qui ne peuvent pas être rattachés à une racine verbale, Cette catégorie inclue aussi les noms propres, les noms communs. Et la deuxième catégorie sont les dérivés qui représente les noms formés à partir d'une racine verbale. Nous trouvons dans cette catégorie les participes actifs (ضَاربٌ – celui qui frappe), les participes passif (مضروب – frappé), les noms de lieux ou de temps (مَضْر بُ – lieu de frappe), le nom d'instrument (مِضْرَبٌ – raquette), le nom d'une fois (ضربة – une frappe), etc.

Le nom peut être :

- Divisé en deux catégories de genre
  - Le masculin – اَلْمُذَكَّر
  - Le féminin – اَلْمُؤَنَّثُ nous ajoutons le ة dans le cas singulier exemple : طويل devient طويلة.
- Divisé en trois catégories de nombre
  - Le singulier – الْمُفْرَد .
  - le duel –الْمُثَنَّى nous rajoutons les deux lettres ان Par exemple : المعلم L'enseignant devient – المعلمان.
  - Le pluriel – اَلْجَمْعُ Pour le masculin nous rajoutons les deux lettres ين où ون dépendamment de la position du mot dans la phrase, exemple : المحب l'amoureux devient المحبين ou : المحبون les amoureux.

Peut être défini – اَلْمَعْرِفَة ou indéfini – اَلنَّكِرَة .

Le nom défini est divisé en 6 catégories :

- Le nom propre – اَلْعَلَمُ;
- Les pronoms – اَلضَّمَائِرُ;
- Les pronoms démonstratifs – أَسْمَاءُ الإِشَارَةِ;
- Les pronoms relatifs – اَلأَسْمَاءُ الْمَوْصُولَة ;
- Le nom défini par – المُعَرَّفُة بِالأَلِفْ وَ اللاَّمْ;
- L'annexé dans le cas de l'annexion – اَلْمُضَافُ إِلَى مَعْرِفَة.

- **verbe**

Est une entité portant un sens dépendant du temps et qui exprime une action, ou un événement, La plupart des mots en arabe, dérivent d'un verbe de trois lettres. Chaque verbe est donc la racine d'une famille de mots.

La conjugaison des verbes se fait par l'ajout des préfixes ou des suffixes à la racines, la forme de base des verbes qu'équivaut en français l'infinitif est sa forme conjuguée à la troisième personne masculine singulier de l'accompli.

- **Les particules**

Les particules grammaticales sont des mots qui n'ont aucune valeur lexicale lorsqu'ils sont pris séparément, mais lorsqu'ils sont associés à d'autres mots ils permettent d'indiquer certains traits grammaticaux tels que le temps le cas où le mode.

Elles jouent également un rôle clé dans la cohérence et l'enchainement d'un texte car ils sont des outils de conjonction de coordination et de subordination à cause de leur fonctionnalité, elles jouent un rôle important dans l'interprétation de la phrase.

Elles sont classées selon leur sémantique et leur fonction dans la phrase en plusieurs types (introduction, explication, conséquence,etc).

nous pouvons distinguer plusieurs types :

- Particules conditionnelles, exemple : (اذما،من،كيفما).
- Particules de coordination, exemple : (و,ف,ثم,أو).
- Particules interrogatives, exemple : (ما,أ,ها)
- Particules relatives, exemple : (ما).
- Particules préposition, exemples : (حتى,من,الى,على,etc).

- Particules de négation, exemple : (لم,لن,لا).
- Particules d'affirmation, exemple : (بلى,أجل,نعم) .
- Particules distinctive, exemple : (أي).
- Particules de futur, exemple : .(س,أن,لن) [28]

La morphologie de la langue arabe permet à un mot peut de véhiculer des informations importantes. Comme un espace délimité symbolique, il révèle plusieurs aspects morphologiques : dérivation, flexion et agglutination [33].

2.3.3 Morphologie dérivée

La morphologie dérivée est le mécanisme de création d'un nouveau mot basé sur un mot existant avec une partie du discours éventuellement différente, Comme les autres langues sémitiques, la morphologie arabe consiste en une représentation racine-et-motif. Tous les mots arabes sont basés sur une « racine », qui est une séquence de consonnes qui contiennent la base Sens du mot [33] Les voyelles et les consonnes sans racine sont ajoutées en suivant des modèles spécifiques pour créer une variété de mots comme l'exemple dans [Tableau 2-3]

Tableau 2-3 – Exemple de significations des trois lettres « ch'r »

1ère itération		2ème itération		3ème itération		4ème itération	
شعر	Sentir	شاعر	Poète	شعر	Poème	شعر	Cheveux

2.3.4 Morphologie flexionnelle

Une morphologie flexionnelle est un changement de la forme des unités lexicales en fonction de facteurs grammaticaux, nous distinguons généralement deux types de flexion : flexion des verbes qui est basé sur la conjugaison des verbes et flexion des noms basé sur la décolonisation des noms.

- **Flexion des verbes**

Un verbe est une entité exprimant un sens dépendant du temps. La majorité des verbes arabes sont formés sur des radicaux de 3 consonnes tel est le cas du verbe « كتب» (kataba– écrire) et éventuellement 4 consonnes tel est le cas du verbe « دحرج» (dahrağa – glisser, faire glisser) [34].

Cette flexion verbale dépend de plusieurs facteurs :

- Le temps (accompli, inaccompli) ;

- Le nombre du sujet (singulier, duel, pluriel) ;
- Le genre du sujet (masculin, féminin) ;
- La personne (première, deuxième et troisième) ;
- Le mode (actif, passif).

Pour les pronoms personnels, le sujet est inclus dans le verbe conjugué. Il n'est donc pas nécessaire de précéder le verbe conjugué par son pronom [35]**.**

- **Flexion des noms**

La décolonisation pour le système nominal est une catégorie de flexion nominale : les noms, les adjectifs et les pronoms. Cette classe concerne les changements sur les noms selon le genre, le cas où le nombre [29] avec un mot ou une expression de la phrase, elle change avec le changement de cette relation sans perdre son sens linguistique [34].

La flexion des noms arabes justifiée par la décolonisation nominale et qui comporte trois cas différents : la nominative " الرفع " (arrafaâ), l'accusative " النصب " (annasub) et la génitive "الجر " (aljaàr) [29].

2.3.5 Morphologie agglutinante

L'arabe est une langue agglutinante, ce qui signifie que le mot peut être attaché un ensemble de clitiques (affixes). Ces clitiques sont divisés en 4 classes [table06] et s'appliquent à une base de mots de manière stricte ordre [Tableau 2-4] :

Tableau 2-4 – Structure du mot arabe

Enclitique	Suffixe	Corps schématique	Préfixe	Proclitique

Par exemple dans le [Tableau 2-5] l'expression arabe »وبحبه« correspond à la forme française « et avec son travail » se divisé en quatre parties (و + ب + حب + ه) :

Tableau 2-5 – Exemple du mot " وبحبه "structuré selon l'ordre

La conjonction		**La particule proclitique**		**La racine/base**		**Le pronom possessif**	
و	Et	ب	Avec	حب	Aimer	ه	Son

Tableau 2-6 – Exemple des affixes

Les conjonctions	و+	“w”	Et
	ف+	“f”	Donc
Les prépositions Proclitique	ل+	“l”	Pour
	ب+	“b”	Avec
	ك+	“k”	Comme
	س+	“s”	Sera
Identifiant	ال+	“al”	Le/la
Les pronoms possessifs	ه+	“H"	Lui/il
	ها+	“Ha: ”	Elle
	هم+	“Hum"	Eux/ils
	هما+	“Huma ”	Eux/elles (deux personnes féminines
	هن+	“Hunna"	Elles (pluriel féminin)
	ك+	“k”	Toi/la tienne
	كم+	“kum”	Vous (pluriel masculin)
	كما +	“kuma”	Vous (deux personne)
	كن+	“kunna”	Vous (pluriel féminin)
	نا+	“na”	Nous
	ي+	“y”	Moi

Différents Préfixe, suffixe et l'apposition génèrent un mot différent à partir de la même racine, la complexité de la structure du mot arabe est l'un des principales difficultés auxquelles les chercheurs sont confrontés lorsqu'ils traitent du sentiment arabe.

2.4 Les problèmes du traitement automatique de la langue arabe

Le traitement automatique des langues c’est le domaine qui s’intéresse à intégrer le langage humain à la machine. Vu à la complexité de la langue arabe l’application de TAL à eux plusieurs problèmes :

2.4.1 L'absence de voyelles

L'absence de voyelles en langue arabe présente un grand problème dans le traitement automatique, en générant plusieurs phénomènes morphosyntaxique et sémantique tel que l'ambiguïté.

En absence de diacritiques un mot arabe peut avoir différentes prononciations avec des différentes significations sans aucun effet orthographique.

Ex : Beauté جمال /Jamalon/

Des chameaux جمال /Jimalon/

2.4.2 Irrégularité de l'ordre des mots dans la phrase

En arabe, l'ordre des mots dans une phrase donnée est libre ce qui génère une flexibilité des phrases. En conséquent, plusieurs règles de combinaison doivent être considérées afin de faire face au problème d'ambiguïtés provoqués par cette flexibilité.

Ex :

- Verbe + sujet + complément :

  (انتشر المرض في العالم) La maladie s'est propagée dans le monde entier.

- Sujet + verbe + complément :

  (المرض انتشر في العالم) C'est la maladie qui s'est propagée dans le monde entier.

- Complément + verbe + sujet :

  (في العالم انتشر المرض) C'est dans le monde entier que la maladie s'est propagée.

2.4.3 Agglutination

L'un des problèmes caractérisé par l'absence des voyelles courtes dans la plupart des textes arabe écrits est l'agglutination qui propose une difficulté en traitement automatique de la langue arabe en causant une augmentation taux d'ambiguïté car les mots peuvent être formés à partir d'une base à laquelle nous pouvons rajouter des affixes (préfixes et/ou suffixes) et des clitiques (enclitiques et/ou proclitiques), et en peut trouver la même unité lexicale en plusieurs découpages possibles à base de leur structure morphologique.

EX : [(et par son travail), (ه =a lui|عمل = travail|بِـ = par|و = et), (ه | عمل | بِ |و), (وبعمله)]

[أَذْهَبُ= Je vais // est-il allé ? = أَذَهَبَ = ذَهَبَ |أ]

2.4.4 Analyse morphologique

L'analyse morphologique est une phase importante dans le traitement automatique. Son objectif principal est de décomposer les mots en morphèmes et d'associer à chaque morphème une information morphologique telle que tag, racine, POS (Part Of Speech) et affixe. Comme nous l'avons vu dans la section précédente.

L'arabe est une langue morphologiquement complexe, Cette complexité nécessite le développement de systèmes appropriés capables de gérer la tokenization, la vérification orthographique, le stemming, la lemmatisation, l'appariement de motifs et le balisage des parties du discours.

De nos jours, de nombreux analyseurs morphologiques pour l'arabe sont déjà développé. Cependant, les systèmes souffrent de limitations importantes, en particulier dans la manipulation d'ambiguïté qui peut résulter de l'omission de signes diacritiques (voyelles courtes), la nature de l'ordre des mots de la phrase arabe, ou la présence de pronom personnel elliptique (ه,ة).

2.4.5 Arabe dialectal

À des fins de communication, les arabophones utilisent généralement l’arabe plutôt que MSA. Il y a environ 30 grands arabes dialectes qui diffèrent du MSA et les uns des autres phonologiquement, morphologiquement et lexicalement. De plus, les dialectes arabes n'ont pas d’orthographes standard et pas d'académie de langues. Par conséquent, en utilisant des outils et des ressources conçus pour MSA pour traiter Les dialectes arabes génèrent des performances considérablement faibles. Récemment, les chercheurs ont commencé à développer des analyseurs syntaxiques pour des dialectes spécifiques comme CALIMA [36] pour le dialecte égyptien. Cependant, ces analyseurs ont encore une faible précision et ne sont faits que pour des dialectes particuliers.

Combler cette lacune dans le traitement de l'arabe améliorera l'information, l'efficacité de la récupération spécifiquement pour les données des médias sociaux.

2.4.6 Arabe romanisé

Arabizi, Arabish ou arabe romanisé fait référence à un système d'écriture arabe utilisant des caractères latins. Il est largement utilisé pour écrire MSA sur les plateformes de

médias sociaux. Faire face à cela forme d'écriture n'a fait l'objet que d'études visant à détecter et convertir Arabizi en arabe.

En ce qui concerne l'Analyse des sentiments, les travaux publiés n'ont pas traité ce problème, car les textes sont prétraités pour filtrer toutes les lettres latines. A notre connaissance, [37] Est le seul travail publié qui ont traité cette tâche.

2.4.7 Reconnaissance d'entité nommée

En arabe, une grande partie des noms arabes sont associés à un adjectif positif. Par exemple, le prénom « سعيد» correspond à l'adjectif qui signifie "heureux". De plus, les noms propres arabes ne sont pas en majuscule comme dans les langues latines, ce qui complique l'identification des entités nommées. Pour cette raison, un système de reconnaissance d'entités nommées (des noms propres) est crucial dans l'analyse des textes arabe et la distinction entre les noms d'entités et les sentiments.

2.5 Prétraitement

En TALN, les données d'entrée dans leur format brut natif peuvent contenir de nombreux mots vides inutiles ou peuvent être mal formaté ce qui poser des problèmes lors de l'analyse. Donc, il est nécessaire de définir et d'appliquer quelques prétraitements sur le corpus de texte avant l'application du traitement nécessaire afin de structurer et nettoyer le texte d'entrée et éliminer le bruit et les données inutiles et extraire les données importantes seulement [38]. Cette étape nommée « Le prétraitement » et c'est l'une des étapes les plus importantes qui précèdent le processus d'apprentissage. La tâche de prétraitement comprend plusieurs étapes [38], [39].

2.5.1 Normalisation

La production d'une forme standard et cohérente de mots à travers plusieurs étapes déférentes en appliquant simple modification dans l'écriture qui n'influe pas considérablement sur le sens du mot afin d'éviter la mal écriture et facilité la manipulation [Tableau 2-7].

Tableau 2-7 – Exemples de normalisation

L'étape	Avant	Apres
Normalisation des lettres répétées	جمييييل	جميل
Suppression de "ال" au début des mots	العربية	عربية

Remplacement de la lettre "ة" par "ه"	عربية	عربيه
Remplacer la lettre "ى" par "ي"	أحلى	احلي
Remplacer les lettres "آ-إ-أ" par "ا"	إنسان	انسان
Allongement	العربيـــــة	العربية
Suppression des signes diacritiques	العَرَبِيَّةُ	العربية

2.5.2 Ajout de balises

Cette étape consiste à remplacés les symboles telles que le point d'exclamation ("!") et le point d'interrogation ("?"), ou bien les symboles émoticônes utilisées dans les réseaux sociaux par mots-clés qui signifier les sentiments représentés par ces symboles [Tableau 2-8].

Tableau 2-8 – Exemple de balises des symboles

!	Point d'exclamation
?	Point d'interrogation
«: (» / « L » / « : - (»	Triste
« :) » / « J » / « :-) »	Heureux

2.5.3 Nettoyage des données

Suppression d'éléments qui n'incluent aucun sentiment et qui peuvent engendrer un bruit dans le traitement :

- Suppression de caractères : /, +, =, *, %, ect ;
- Suppression de nombres ;
- Suppression des balises XML ;
- Suppression des mots et les caractères non arabes.

2.5.4 Tokenization

Cette étape consiste à découper le texte en une séquence de jetons où chaque jeton représente un seul mot, séparés par des espaces blancs ou des caractères de ponctuation et chaque jeton représente un seul mot.

2.5.5 Suppression des mots vides

La suppression de tous les mots vides, inutiles qui n’ont aucun effet ou valeur ajoutée sur le processus de classification.

La liste de mots vides arabes composée des : Adverbes, Pronoms conditionnels, Pronoms interrogatifs Prépositions, Pronoms relatifs, Pronoms. Pronoms verbaux.

2.5.6 Stemming

Stemming est l'une des étapes les plus difficiles, elle nous montre que l'arabe est une langue inflexible, cela est dû aux difficultés existent dans cette langue et qui ont été mentionnées dans ce chapitre. Il s'agit de trouver la racine lexicale ou stem du mot en extrairont les affixes (tels que les infixes, les préfixes et les suffixes) d'un mot [Table-09] en utilisant une liste de suffixes et une autre de préfixes. Pour cela elle travaille par des techniques de racinisation (Larkey Connell [2002] (light10), A ljlayl Frieder [2002], Darwish Orad [2003], Chen Gey [2002], Kadri Nie [2008], etc).

Les algorithmes de stemming arabe sont classés en trois catégories :

- Approche basée sur la racine (Khoja Stemmer) ;
- Approche basée sur stem (Larkey Light Stemmer) ;
- Approche statistique (implique souvent N-Gram).

Tableau 2-9 – Exemple de lemmatisation

Post fixe	Suffixe	Racine	Préfixe	Antéfixe
ه	ون	توقع	ت	أ
Pronom suffixe complément du nom	Suffixe verbal exprimant le pluriel	Dérivé de la racine	Verbal du temps de l'inaccompli	Conjonction d'interrogation

- **Stemmer de Khoja**

Ce lemmatiseur a le rôle de supprimer les plus longs suffixes et préfixes. Il compare ensuite le mot restant avec des motifs verbaux et nominaux pour l'extraction de la racine. Pour ce faire, il fait appel à plusieurs fichiers de données linguistiques comme une liste de tous les caractères diacritiques, des articles précis, caractères de ponctuation, et 168 mots fonctionnels.

Le lemmatiseur de khoja traite plusieurs difficultés comme :

- Si la racine contient de longues voyelles (alif, waw, Ya) (ا، و، ي) la forme de cette lettre peut changer durant la dérivation ;

- Si la racine contient « hamza » (ء), ce hamza peut changer sa forme durant la dérivation, l'analyseur détectera ça et retournera sa forme originale ;
- L'analyseur ne donne aucun résultat en cas des mots qui non pas des racines tel que les pronoms personnels ;
- La lettre de racine peut être éliminée durant la dérivation. L'analyseur tente de détecter la lettre pour reconstituer la bonne racine.

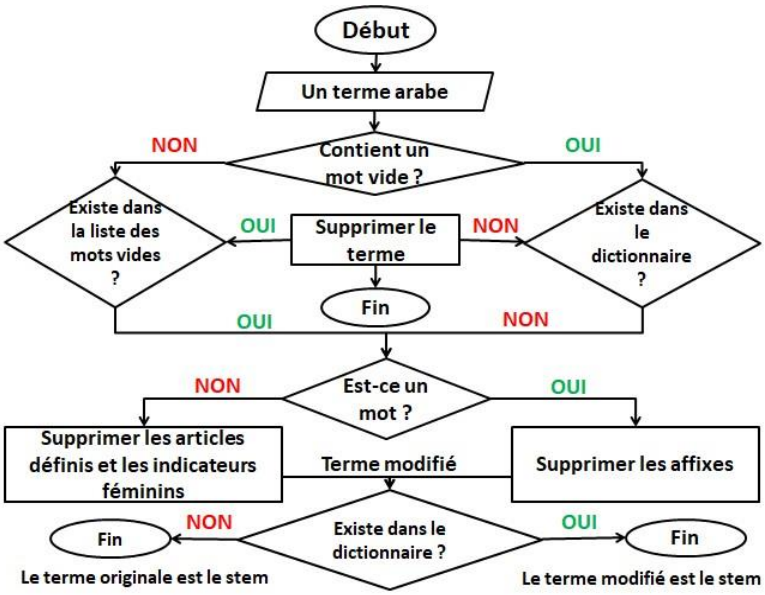

Figure 2-1 – Organigramme lemmatiseur de khoja

- **Stemmer d'ISRI**

ISRI (The Information Science Research Institute), est un lemmatiseur léger arabe sans dictionnaire des racines. Il utilise une liste étendue d'affixes [Tableau 2-10] avec des schémas de dérivation des plus fréquents [Tableau 2-11] pour extraire des racines. L'algorithme de ce lemmatiseur

- Supprime les signes diacritiques qui représentent les voyelles ;
- Normalisation de Hamza (changer les lettres ء ،ؤ et ئ Par la lettre أ) ;
- Supprimez les préfixes à deux et trois lettres du mot dans cet ordre ;
- Supprimez les lettres de liaison و S'il y a une lettre و Dans le préfixe du mot ;
- Normalisation d'Alif (Changement des lettres (إ، أ، آ) par (ا) ;
- Stemmer renvoie la même lettre si le mot est à 3 lettres. Et retourner le même mot si le mot est ambigu.

Considérez 4 cas selon la longueur du mot :

1. Longueur de 4 lettres : Extraction de la racine appropriée si la forme du mot correspondant aux schèmes de forme PR4, sinon par suppression des suffixes et des préfixes de longueur 1 de S1 et P1 dans cet ordre, et retournement de la racine si le mot a au moins trois lettres.

2. Longueur de 5 lettres : Extraction de la racine trilitère du mot si la forme du mot correspondant aux schèmes de forme PR5. Sinon, nous retirons les suffixes et les préfixes et nous retournons la racine trilittère. Si la longueur du mot est encore de cinq caractères, le comparer avec les schèmes de PR54 et retourner la racine dans le cas où elle a une longueur de 4 lettres.

3. Longueur de 6 lettres : Extraction de la racine trilitère du mot si la forme du mot correspondant aux schèmes de forme PR63, sinon suppression des suffixes. Si la suppression d'un suffixe résulte en un terme avec cinq caractères, alors renvoyer ce terme à l'étape 2. Dans le cas où ce n'est toujours pas satisfaisant, nous supprimons alors les préfixes de longueur 1. En cas de succès, nous renvoyons le terme à l'étape 2. Si le mot a toujours six caractères, nous le comparerons avec les schèmes de PR64 et nous retournerons sa racine dans le cas où elle a une longueur de 4 lettres.

4. Longueur de 7 lettres : Suppression des suffixes de longueur 1 et envoie du terme résultant à l'étape précédente. Dans le cas où cela ne marchera pas, alors nous supprimerons les préfixes de longueur 1 et nous renverrons le terme obtenu à l'étape précédente.

Tableau 2-10 – Les antéfixes proposées par ISRI [40]

Le type de l'ensemble		**Description**	**Description**
Diacritiques		Les diacritiques de vocalisation	ّ-ْ-ٌ-ُ-ٍ-ِ-ً-َ
Préfixes	P3	Les préfixes de longueur 3	ولل وال كال ال
	P2	Les préfixes de longueur 2	لل ال
	P1	Les préfixes de longueur 1	ا، ن، ت، ي، ف، و، س، ب، ل
Suffixes	S1	Les suffixes de longueur 3	كمل، تين، تان، همل، تمل
	S2	Les suffixes de longueur 2	هم، ما، وا , ني، تن، تم، ها، يا نا, هن، كم، تن، ين، ان, ات، ون
	S3	Les suffixes de longueur 1	ن, ا , ت, ك , ي, ه, ة

Tableau 2-11 – Les schèmes et leurs racines proposées par ISRI [40]

Le type de l'ensemble	Description	Leur contenu proposé
PR4	Les schèmes de longueur 4	فاعل فعول فعلة فعال فعيل مفعل
PR53	Les schèmes de longueur 5 et racine de longueur 3	تفاعل افتعل افعال افاعل فعالة فعالن فعولة تفعلة تفعيل مفعلة مفعول فاعول فواعل مفعال مفعيلل افعلة فعائل منفعل مفتعل فاعلة مفاعل فعالع يفتعل تفتعل فعالي انفعل
PR54	Les schèmes de longueur 5 et racine de longueur 4	تفعلل افعلل مفعلل فعللة فعالن فعالل
PR63	Les schèmes de longueur 6 et racine de longueur 3	استعفل مفعالة افتعال افعوعل انفعال مستفعل
PR64	Les schèmes de longueur 6 et racine de longueur 4	افتعلل افعالل متفعلل

2.6 L'apprentissage automatique

Dans le domaine de l'analyse des sentiments et la catégorisation de texte les chercheurs s'évertue depuis des années à développer des programmes afin d'augmenter la vitesse et la consistance d'analyse. Les programmes travaillent bien plus rapidement et plus méthodiquement dans ce domaine contrairement aux humains. L'apprentissage automatique est le meilleure outille pour atteindre ce but. Il existe principalement deux types d'algorithmes d'apprentissage catégorisés selon leur mode d'apprentissage : l'apprentissage supervisé requiert le plus d'interaction humaine, car il implique que le système reçoive un ensemble des données bien défini déjà analysées, annotées et classées par des experts pour permet au system de déceler des modèles au sein des données et de les appliquer à un processus qui analyser et étiqueter des données brutes, en comparant sa sortie réelle avec les sorties enseignées pour trouver des erreurs et modifier le modèle par adaptation ce qui aide à la détection de polarité « positif » ou « négatif » [41].

Au contraire, l'apprentissage non supervisé est plus complexe, les données sont non labellisées, il est utilisé lorsque le problème nécessite une quantité massive de données non étiquetées. A l'aide de son algorithme le system trouve tout seul les cas de similarités parmi ses données d'entrée pour les classer et analyser en fonction des tendances ou des clusters qu'ils décèlent données sans aucune intervention humaine.

2.7 Algorithmes de classification

La classification est le processus de reconnaissance, de compréhension et de regroupement d'idées et d'objets pour catégorise les nouvelles données en un nombre distinct de classes et des étiquettes sont attribuées à chaque classe, Il dépend une variété d'algorithmes effectuée sur des données structurées ou non structurées pour les analyser et affecter en fonction de leurs caractéristiques ou attributs, à telle catégorie ou telle classe prédéfinie laquelle une nouvelle donnée appartiendra.

Il existe plusieurs algorithmes de classification chacun est utilisé pour résoudre un problème spécifique telle que.

2.7.1 Support Vector Machines (SVM)

Les classificateurs Support Vector Machine (SVM) sont des classificateurs binaires qui appartient à la catégorie des classificateurs linéaires mais n'ont pas limités à devenir linéaire seulement car ils sont préférés à tout modèle de classification en raison de leur fonction de noyau, ce qui améliore l'efficacité des calculs. Cet algorithme joue un rôle essentiel dans les problèmes de classification et plus généralement, dans les algorithmes supervisés d'apprentissage automatique, il représente chaque élément de données d'apprentissage comme un point dans un espace séparés en catégories à n dimensions (où n est un nombre de caractéristiques que vous avez) avec la valeur de chaque caractéristique étant la valeur d'une coordonnée particulière. Ensuite, de nouveaux exemples sont cartographiés dans ce même espace et prédits comme appartenant à une catégorie en fonction de quel côté de l'écart ils se situent. Pour ce but, son processus consiste à trouver des limites de décision qui classent les points de données connu sous le nom de « Les hyperplans » [42].

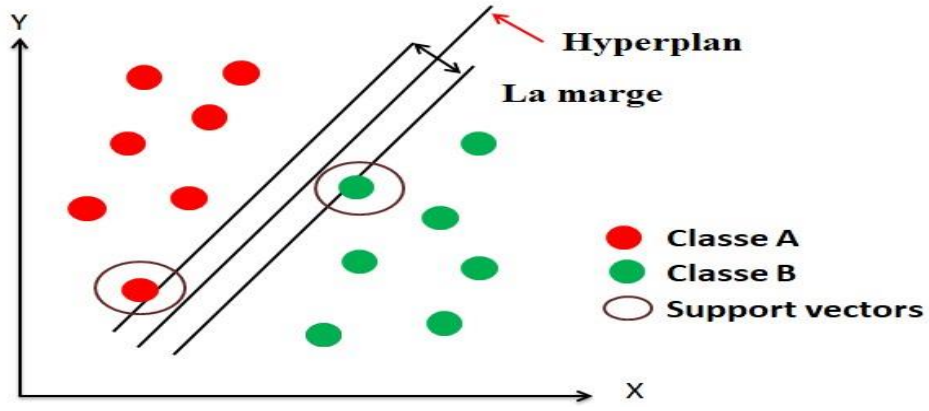

Figure 2-2 – Exemple de classification par SVM

2.7.2 K Nearest Neighbors (KNN)

K Nearest Neighbors est un simple algorithme d'apprentissage automatique sans aucune hypothèse qui ne tente pas de construire un modèle interne général, mais stocke et mémorise simplement tous les instances des données d'apprentissage disponibles [43], puis classer un point (nouvelle texte donnée) en fonction de la classe de ses K voisins les plus proches (ensemble des textes du jeu d'apprentissage qui lui est plus proche) à l'aide d'une fonction de distance pour mesurer.

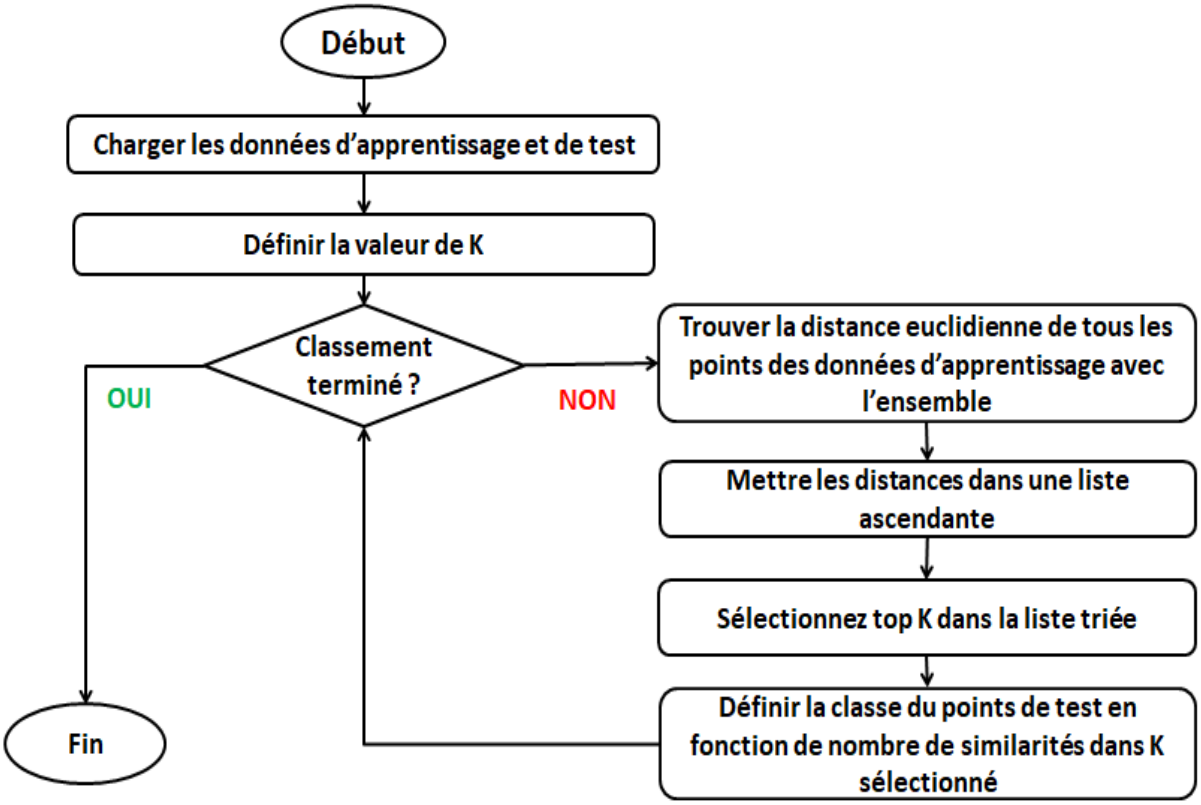

Figure 2-3 – Organigramme d'algorithme K-NN

2.7.3 Random Forest (RF)

Random forest (RF) est un algorithme de classification supervisée qui a été développé par Breiman et Cutler en 2001. Il s'agit d'une approche d'ensemble pour la classification et la régression qui fonctionne en construisant divers arbres de décision au moment de l'entrainement [44]. La prédiction ou classification se fait alors en fonction d'un système de

vote majoritaire au sein de ces différents arbres. Le principe de RF est alors de chercher à tirer profit de cette instabilité en les agrégeant entre eux.

Random forest se construit en concevant un arbre sur des sous-échantillons tirés au hasard. Ensuite, pour chacun des arbres à construire, un sous-ensemble de q ~ p est sélectionné aléatoirement et sert à leur élaboration respective. L'objectif de cette approche est de rendre les arbres construits plus indépendants entre eux ce qui offre de meilleures performances lors de l'agrégation en forêt.

Cette approche a l'avantage d'être applicable à des données de grande dimension et d'être simple à mettre en œuvre. L'utilisation de RF élimine également toute phase d'élagage et de tout problème lié à la multi colinéarité des variables.

Les RFs sont devenues populaires ces dernières années, car les performances de ce type d'algorithmes sont exceptionnelles pour les tâches de classification dans certains domaines tels que la bioinformatique et la biologie computationnelle.

2.8 Conclusion

Dans ce chapitre nous avons fait une recherche sur les caractéristiques et la morphologie arabe, la difficulté du traitement automatique de cette sémitique et flexionnelle langue, et les différentes étapes de prétraitements de données qui doivent être réalisés avant le traitement nécessaire. Ensuite nous avons étudié l'apprentissage automatique avec ces types et leur application en TALN, en finir par quelques algorithmes de classification et les types de classificateurs utilisée en traitement des sentiments des langues.

3 Chapitre 3 Les algorithmes bio-inspirés

3.1 Introduction

Avec l'incompatibilité des techniques existantes, il a été nécessaire pour les scientifiques d'identifier de nouvelles méthodes qui s'adapte précisément aux problèmes posés en informatique, et la diversité des phénomènes dans la nature a été une riche source d'inspiration pour eux afin de réaliser ce problème.

Ces algorithmes s'inspirent en particulier de l'évolution naturelle comme les algorithmes génétiques et l'évolution différentielle. En plus, un nouveau comportement d'inspiration est introduit récemment à base d'intelligence distribuée (Swarm Intelligence) qui cherche à imiter le comportement collectif des espèces tel que les insectes et les animaux tel que l'algorithme Grey wolf optilizer et l'algorithme des faucons. Ils commencent le processus d'optimisation en générant un ensemble d'individus, où chaque individu de la population représente une solution La population évoluera de manière itérative en remplaçant la population actuelle par une population nouvellement générée en utilisant des opérateurs souvent stochastiques. Le processus d'optimisation est poursuivi jusqu'à ce qu'un critère d'arrêt soit satisfait.

3.2 Les algorithmes évolutionnaires

Les algorithmes évolutionnaires (Evolutionary Algorithms ou Evolutionary Computation) font partie, d'un point de vue informatique, de la famille des algorithmes d'optimisation stochastiques inspirés du paradigme de l'évolution darwinienne des espèces. Le but d'un AE est dévaluer un ensemble de solutions appelées population vers une solution optimale en créant une population initiale (une collection d'individus).

Après l'évaluation de cette population, puis son évolution à travers plusieurs générations, seuls les individus les plus aptes, à savoir, ceux qui représentent la meilleure solution de la population sont conservés et sont autorisés à croiser avec d'autres membres aptes. Et puis l'effectuation du croisement, afin de créer des individus qui sont plus en forme que les deux parents en prenant les meilleures caractéristiques de chacun des parents, ce qui fait converger les solutions (individus) vers l'optimum.

3.2.1 Algorithme génétique (GA)

L'algorithme génétique est inspiré de la théorie de l'évolution et des règles de la génétique. En biologie, chaque individu a un chromosome unique dans lequel le code génétique de cet individu est sauvegardé. Et pour les algorithmes génétiques les points de l'espace de recherche sont représentés par ces chromosomes codés généralement par des chaines de bits.

Les GAs ont prouvé leur succès dans les problèmes d'optimisation à large espace de solutions [45]. Ils sont utilisés lorsque la recherche exhaustive d'une solution est couteuse en termes de temps d'exécution.

L'algorithme génétique de base peut être expliqué à l'aide des étapes suivantes :

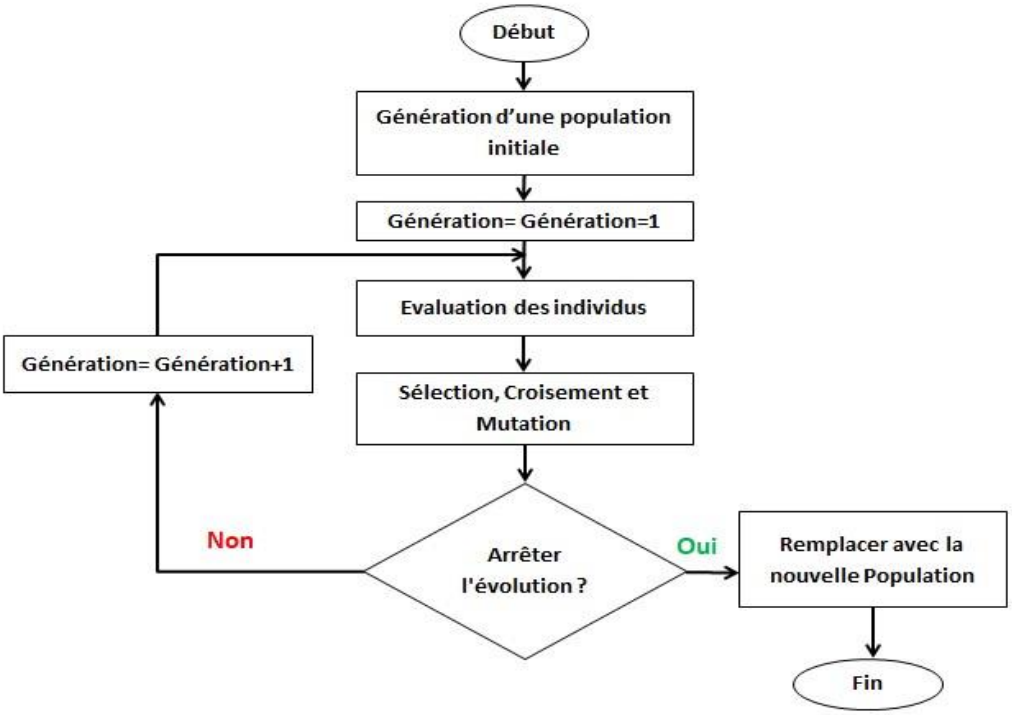

Figure 3-1 – Organigramme d'algorithme génétique

- **Sélection des parents**

Cette opération repose sur le principe d'adaptation de chaque individu de la population à son environnement et suit la théorie de la sélection naturelle proposée par Charles Darwin. Par conséquent, seuls les individus ayant des coûts de performance élevés sont sélectionnés pour survivre et se reproduire. Parmi les méthodes de sélection les plus célèbres on a la

sélection proportionnelle à la fonction fitness, la sélection par la roulette, la sélection sur le rang et la sélection en tournoi.

- **Génération aléatoire de la population initiale**

La vitesse de l'algorithme est affectée par la population initiale d'individus. Si la localisation optimale dans l'espace d'états est inconnue, il est naturel de générer aléatoirement une population d'individus en effectuant des extractions uniformes dans chacun des domaines associés aux composantes de l'espace d'états, tout en s'assurant que les individus produits respectent les contraintes.

Cependant, il est naturel d'engendrer les individus dans un sous-domaine particulier afin d'accélérer la convergence si des informations sur le problème sont déjà disponibles.

- **Gestion des contraintes**

Au cours du processus de sélection, les éléments de la population qui violent les contraintes se verront attribuer une mauvaise fitness et seront probablement éliminés. Ils peuvent être retenus en les pénalisant, car ces éléments inacceptables peuvent permettre la génération d'éléments acceptables de bonne qualité.

Pour de nombreux problèmes, la valeur optimale est atteinte lorsqu'au moins une des contraintes de séparation sature, c'est-à-dire sur les limites de l'espace admissible. La gestion des contraintes en pénalisant la fonction fitness est difficile, et le « dosage » est nécessaire pour ne pas favoriser la recherche d'une solution acceptable au détriment de la recherche de l'optimum, et inversement.

- **Le croisement**

Dans cette opération, de nouvelles solutions sont créées à partir de la population existante pour enrichir la diversité de la population en manipulant la structure des chromosomes. Classiquement, les croisements sont induits avec deux parents et engendrés deux enfants.

Il existe plusieurs techniques de croisement sont:

- Croisement en un point ;
- Croisement multipoint ;
- Croisement uniforme ;
- Croisement aléatoire ;
- Croisement de préservation de la priorité ;
- Croisement de l'ordre de Davi ;

- Croisement commandé ;
- Croisement partiellement adapté.

- **La mutation**

C'est la modification ou la transmission d'une ou plusieurs valeurs génétiques des chromosomes à la génération suivante, pour obtenir une solution peut être entièrement différente de la solution précédente. Cet opérateur nous garantit que l'algorithme génétique sera susceptible d'atteindre tous les points de l'espace d'état, sans pour autant les parcourir tous dans le processus de résolution. Cet opérateur nous garantit que l'algorithme génétique sera susceptible d'atteindre tous les points de l'espace d'état, sans pour autant les parcourir tous dans le processus de résolution.

Les différents opérateurs de mutation sont :

- Interchanger ;
- Déplacement ;
- Insertion ;
- Inversion déplacée ;
- Réinitialisation aléatoire.

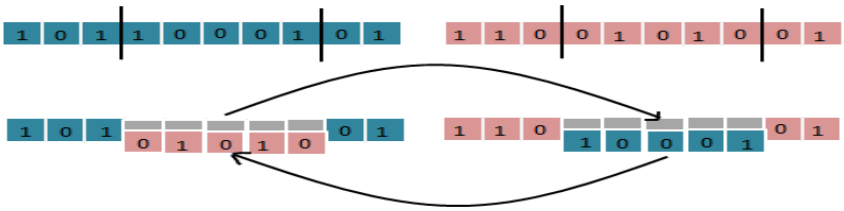

Figure 3-2 – Croisement

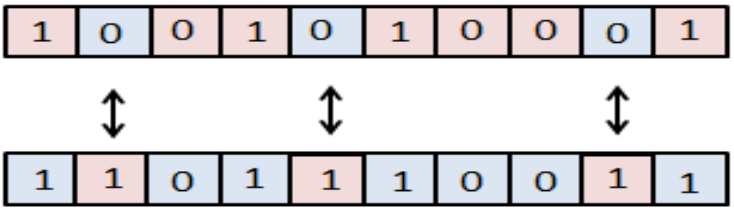

Figure 3-3 – Mutation

3.2.2 Algorithme de l'évolution différentielle (DE)

L'évolution différentielle ("Differential Evolution", DE) est une métaheuristique stochastique d'optimisation proposée par Storn et Price, qui a été inspirée par les algorithmes génétiques et des stratégies évolutionnaires combinées avec une technique géométrique de recherche. DE est un algorithme basé sur une population initiale aléatoire comme les algorithmes génétiques, il utilise les mêmes principes que les GAs « croisement, mutation et sélection » [46].

La différence principale en construisant de meilleures solutions est que les GAs se fondent sur le croisement tandis que le DE se fonde sur l'opération de mutation.

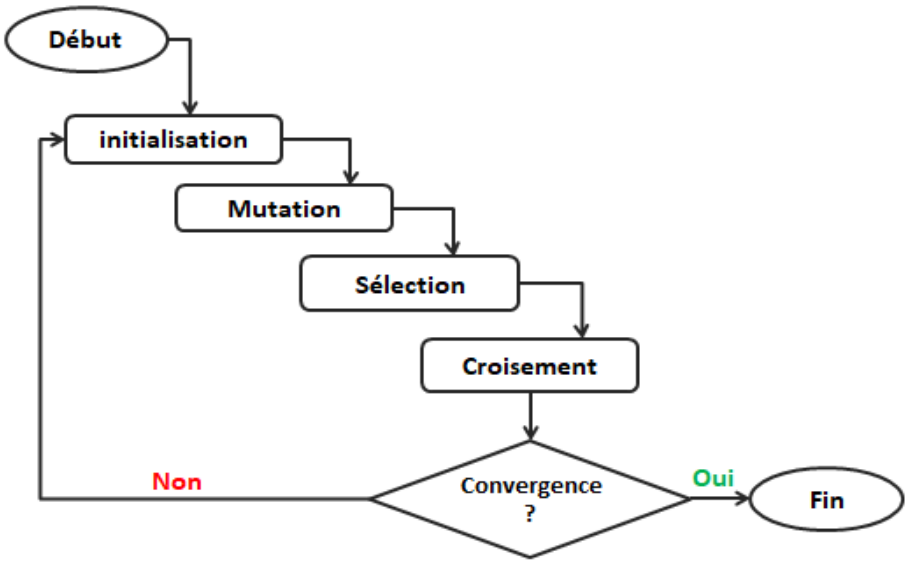

Figure 3-4 – Organigramme d'algorithme DE

Dans la méthode DE, la population initiale est générée par tirage aléatoire uniforme de toutes les valeurs possibles de chaque variable. Les bornes inférieure et supérieure des variables sont spécifiées par l'utilisateur en fonction de la nature du problème. Après l'initialisation, l'algorithme effectue une série de transformations sur les individus dans un processus appelé évolution. Une population contient N individus. Chaque individu $x_{i,G}$est un vecteur de dimension D, où G représente la génération :

$$\boldsymbol{x_{i,G}} = \left(\mathbf{x_{1i,G}}, \mathbf{x_{2i,G}}, \cdots \cdots, \mathbf{x_{Di,G}}\right) \quad \textbf{avec} \quad \mathbf{i} = \{\mathbf{1}, \mathbf{2}, \cdots, \mathbf{N}\} \tag{3-1}$$

Pour chaque génération, l'algorithme DE applique successivement les trois opérations «mutation, croisement et sélection» sur chaque vecteur pour produire un vecteur d'essai (trial vector) :

$$\boldsymbol{u_{i,G+1} = (u_{1i,G+1}, u_{2i,G+1}, \cdots \cdots, u_{Di,G+1}) \text{ avec } i = \{1, 2, \cdots, N\}} \quad (3\text{-}2)$$

Une opération de sélection permet de choisir les individus à conserver pour la nouvelle génération (G + 1).

- **Mutation :**

Pour chaque vecteur courant $x_{i,G}$, nous génèrons un vecteur mutant $v_{i,G+1}$ qui peut être créé en utilisant une des stratégies de mutation suivantes [46] :

DE/rand/1 :

Cette notation indique que le vecteur à perturber est aléatoirement choisi et que la perturbation se compose sur une seule différence.

$$\mathbf{v_{i,G+1} = x_{r1,G} + F(x_{r2,G} - x_{r3,G})} \quad (3\text{-}3)$$

DE/best/1

Comme la stratégie précédente, l'individu de la prochaine génération est produit par le meilleur membre de la population en utilisant la formule :

$$\boldsymbol{v_{i,G+1} = x_{best,G} + F(x_{r1,G} - x_{r2,G})} \quad (3\text{-}4)$$

DE/rand to best/1

Cette stratégie place la perturbation à un endroit entre un membre aléatoirement choisi de la population et le meilleur membre de cette dernière :

$$\boldsymbol{v_{i,G+1} = x_{i,G} + F(x_{r1,G} - x_{r2,G}) + F(x_{best,G} - x_{i,G})} \quad (3\text{-}5)$$

DE/best/2

DE/best/2 emploie deux vecteurs de différence comme perturbation :

$$\mathbf{v_{i,G+1} = x_{best,G} + F(x_{r1,G} - x_{r2,G}) + F(x_{r3,G} - x_{4,G})} \quad (3\text{-}6)$$

DE/rand/2

Cette stratégie est générée par

$$\mathbf{v_{i,G+1} = x_{r1,G} + F(x_{r2,G} - x_{r3,G}) + F(x_{r4,G} - x_{r5,G})} \quad (3\text{-}7)$$

Tels que :

$x_{i,G}$: i^{th} individu de la génération courante G

r1, r2, r3, r4 · · · rN: Entiers aléatoires et tous différents ∈ {1, 2, · · ·, N}

x : Ensemble de population

F : (Differential weight) Constante de mutation ∈ [0,2]

- **Croisement**

L'opération du croisement est appliquée sur les individus par la règle :

$$\mathbf{u_{i,G+1}} = \mathbf{x_{i,G}}.(\mathbf{1} - \mathbf{pc}) + \mathbf{v_{i,G+1}}.\boldsymbol{p_c} \qquad (3\text{-}8)$$

- **Sélection**

Toutes les solutions dans la population ont la même chance d'être sélectionnées comme des parents selon la fonction d'adaptation. L'enfant produit après les opérations de mutation et de croisement est évalué. Puis, la performance de l'enfant et son parent est comparée et le meilleur entre eux est choisi. Si le parent est encore meilleur, il est maintenu dans la population en utilisant la formule suivante:

$$\mathbf{x_{i,G+1}} = \begin{cases} \mathbf{u_{i,G+1}} & \mathbf{f(u_{i,G+1}) < f(x_{i,G})} \\ \mathbf{x_{i,G}} & \mathbf{sinon} \end{cases} \qquad (3\text{-}9)$$

3.3 Les algorithmes inspirés de l'intelligence distribuée

3.3.1 Algorithme des loups gris (GWO)

GWO (Grey Wolf optimisation algorithm) est une méthode d'optimisation métaheuristique, proposée par [47], inspirée de la hiérarchie sociale et des techniques de chasse des loups gris dans la nature. Le loup gris (Canis lupus) appartient à la famille des canidés, il est considéré comme un prédateur apex, ce qui signifie qu'il se situe au sommet de la chaîne alimentaire.

Les loups gris préfèrent vivre en meute dans des groupes de 5 à 12 personnes en moyenne. Il est particulièrement intéressant de noter qu'ils ont pour chaque groupe une hiérarchie sociale dominante très stricte et subdivisée en quatre catégories, qui contienne les loups alpha, bêta, delta et oméga [47], [48].

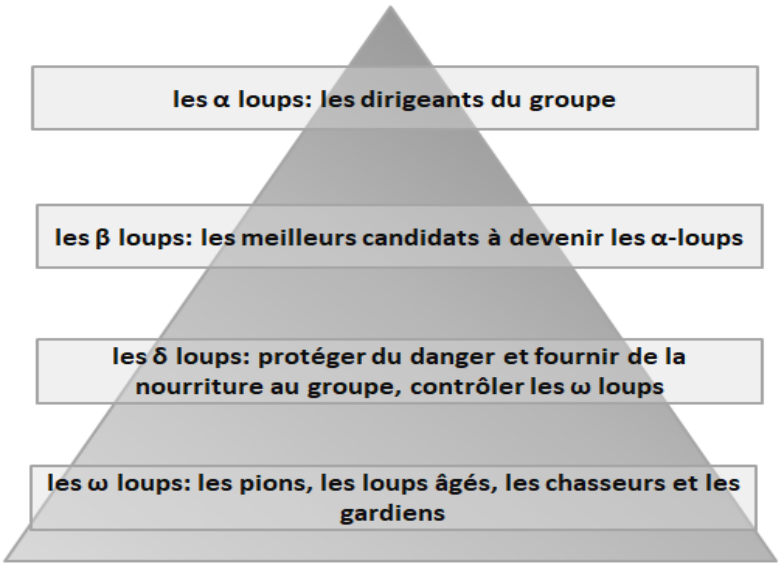

Figure 3-5 – La hiérarchie sociale des loups gris

Le loup alpha α également appelé « loup dominant », qui peut être un mâle ou femelle. Il est principalement responsable des décisions concernant la chasse, le lieu de couchage, l'heure du réveil, etc.

Le loup bêta β est le deuxième niveau de la hiérarchie. Les béta sont des loups subordonnés qui aident l'alpha dans la prise de décision ou dans d'autres activités de la meute. Le loup bêta peut être un mâle ou une femelle, et il est probablement le meilleur candidat pour devenir l'alpha au cas où le loup alpha n'arrive pas à orienter le troupeau, devient très vieux ou décède.

Les loups delta et oméga sont au rang le plus bas dans la meute de loups gris. Les deux sont presque de type égal avec seulement une légère différence sur leur caractère dominant, le delta δ a doivent se soumettre aux alphas et aux bêtas, mais ils dominent les omégas.

L'oméga ω joue le rôle d'un bouc émissaire afin de se soumettre aux autres loups dominants. Cela montre que l'oméga est du moins important dans la meute des loups [47].

- **Hiérarchie sociale**

Pour la modélisation mathématique de la hiérarchie sociale des loups lors de la conception du GWO, nous considérons la solution la plus adaptée comme l'alpha (α), les deuxièmes et troisièmes meilleures solutions sont nommées respectivement bêta (β) et delta (δ). Le reste des solutions candidates est supposé être oméga (ω) [47].

Dans l'algorithme GWO, la chasse (optimisation) est guidée par α, β et δ. Les loups ω suivent ces trois loups.

- **Encerclement de la proie**

Pendant la chasse les loups gris encerclent leurs proies, et afin de modéliser mathématiquement ce comportement d'encerclement, les équations suivantes sont proposées :

$$\boldsymbol{D} = |\boldsymbol{C}.\boldsymbol{X_p}(\boldsymbol{t}) - \boldsymbol{X}(\boldsymbol{t})| \qquad (3\text{-}10)$$

$$\boldsymbol{X}(\boldsymbol{t}+\boldsymbol{1}) = \boldsymbol{X_p}(\boldsymbol{t}) - \boldsymbol{A}.\boldsymbol{D} \qquad (3\text{-}11)$$

Tel que :

t : L'itération en cours

X_p **:** Le vecteur de position de la proie

X : le vecteur de position.

A : vecteurs de coefficients

$$\boldsymbol{A} = \boldsymbol{2a} * \boldsymbol{r_1} - \boldsymbol{a} \qquad (3\text{-}12)$$

C : Des vecteurs de coefficients

$$\boldsymbol{C} = \boldsymbol{2} * \boldsymbol{r_2} \qquad (3\text{-}13)$$

Où a est diminué linéairement de 2 à 0 avec $a = 2 - t * \left(\frac{2}{T}\right)$, tandis que r_1, r_2 sont des valeurs aléatoires dans [0.1], t représente l'itération en cours et T : le nombre maximum d'itération.

- **La chasse**

Les loups gris ont la capacité de reconnaître l'emplacement de leurs proies et de les encercler. La chasse est généralement guidée par l'alpha. Le bêta et le delta peuvent également participer à la chasse de manière occasionnelle. Cependant, dans un espace de recherche abstrait, nous n'avons aucune idée de l'emplacement de l'optimum (la proie)

Pour la simulation mathématique du comportement de chasse des loups gris, nous supposons que les alphas (meilleure solution candidate) bêta et delta ont une meilleure connaissance à propos de l'emplacement potentiel de la proie. Par conséquent, nous sauvegardons les trois meilleures solutions obtenues jusqu'à présent et on met à jour la position des autres loups (y compris les omégas) en fonction de la position des meilleurs agents de recherche [47].

Cette étape est donnée par les équations suivantes :

$$\boldsymbol{D_\alpha = |C_\alpha * X_\alpha (t) - X(t)|} \quad (3\text{-}14)$$

$$\boldsymbol{D_\beta = |C_\beta * X_\beta (t) - X(t)|} \quad (3\text{-}15)$$

$$\boldsymbol{D_\delta = |C_\delta * X_\delta (t) - X(t)|} \quad (3\text{-}16)$$

$$\boldsymbol{X(t+1) = (X_1 + X_2 + X_3)/3} \quad (3\text{-}17)$$

Tel que :

X_α (t) : La position de l'alpha

X_β (t) : La position de la bêta

X_δ(t) : La position de delta

C_1, C_2, C_3 : Des vecteurs aléatoires

X : La position de la solution actuelle.

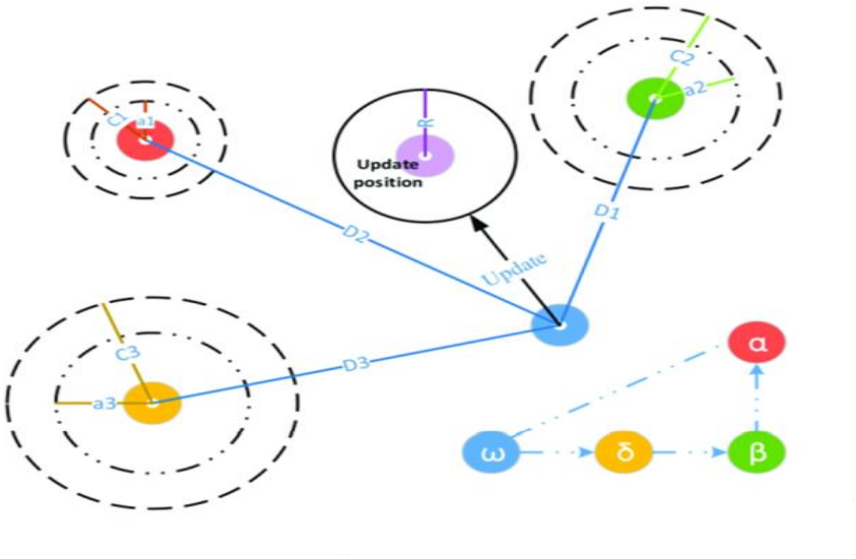

Figure 3-6 – Changement de position [49]

- **L'attaque de la proie**

La dernière étape consiste en l'attaque de la proie lorsque elle cesse de bouger alors le loup gris termine la chasse.

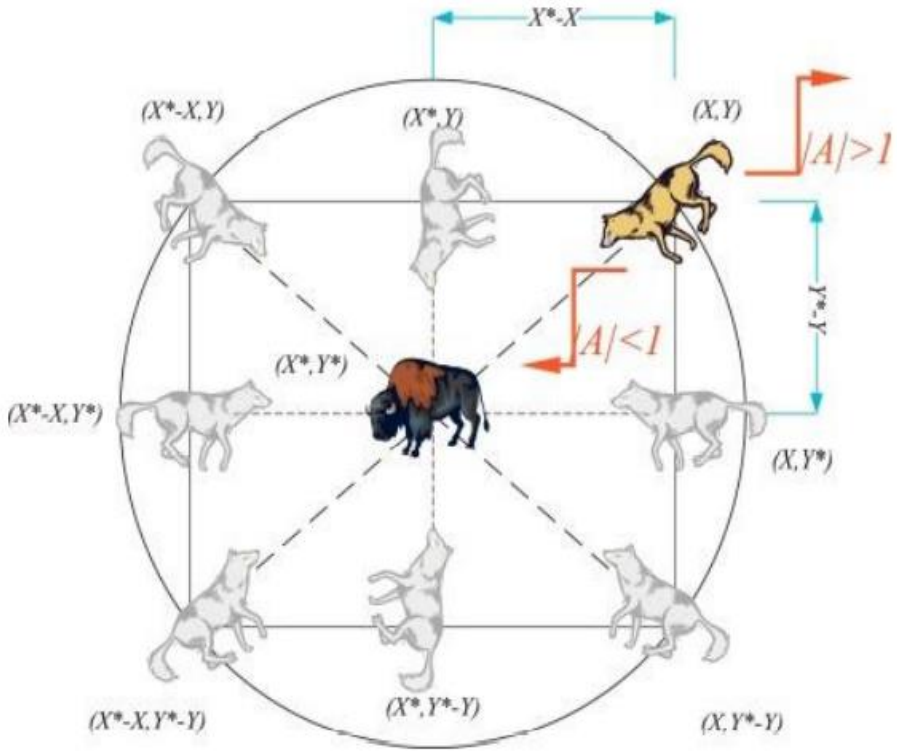

Figure 3-7 – Attaque des loups gris [47]

Durant la chasse, la phase d'exploration est appliquée tant que la condition :

- $|A| \geq 1$ soit respectée : les loups sont forcés à s'éloigner les uns des autres et de diverger de la proie. Ensuite, la phase d'exploitation est entamée ;
- $|A| < 1$: les loups sont alors forcés d'attaquer la proie.

Par conséquent, un paramètre clé nommé « a » qui a pour but d'assurer une balance entre la phase d'exploration et d'exploitation. Ce paramètre « a » décroit linéairement de 2 à 0 durant le processus d'exécution [47], [48].

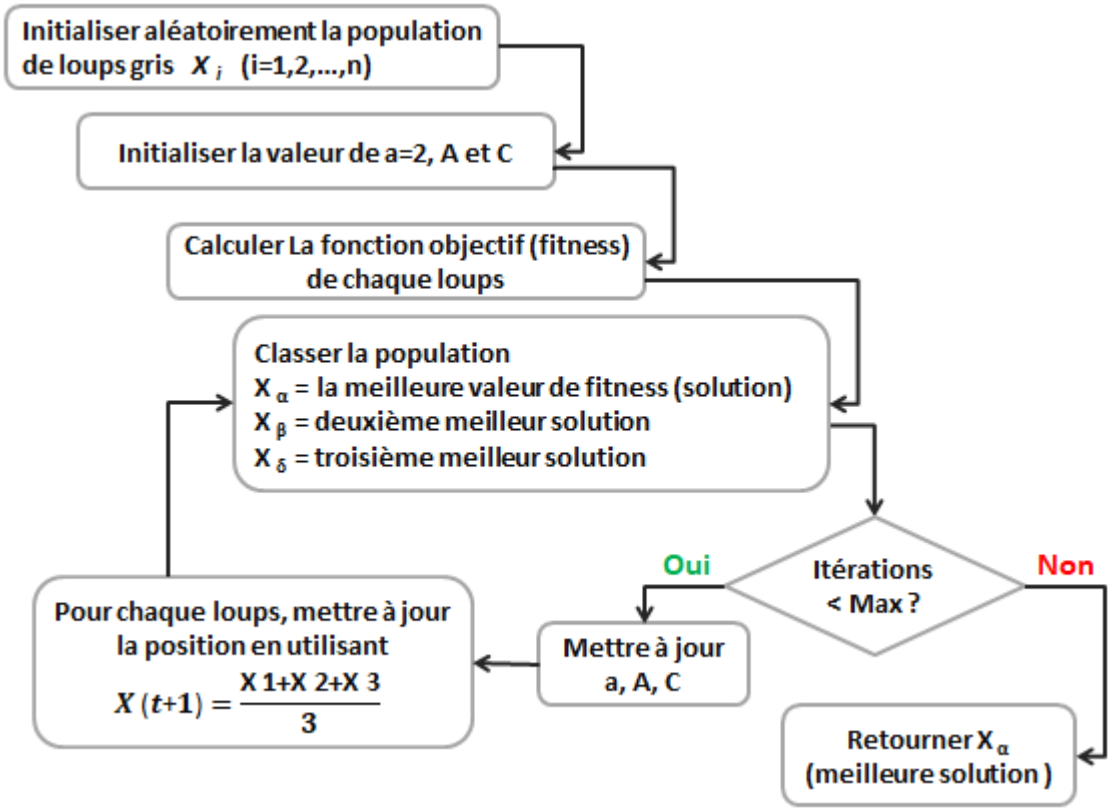

Figure 3-8 – Algorithme de GWO

3.3.2 Algorithme des faucons (HHO)

L'algorithme des faucons (Harris Hawks Optimisation (HHO)) est un algorithme d'intelligence des essaims basé sur la population qui est proposé par [50], et dès le premier jour, il a attiré de plus en plus l'attention des chercheurs en raison de sa structure flexible, de ses hautes performances et de ses résultats de haute qualité. La logique principale de la méthode HHO est conçue sur la base du comportement coopératif et des styles de chasse des faucons de Harris dans la nature appelés «Pounce surprise», également connu sous le nom de stratégie des (sept tueries).

Dans cette stratégie intelligente, plusieurs faucons se lancent en coopération sur une proie provenant de différentes directions pour tenter de la surprendre. Les faucons de Harris peuvent révéler une variété de modèles de poursuite basés sur la nature dynamique des scénarios et les modèles d'évasion de la proie. Ce travail imite mathématiquement les modèles et comportements dynamiques pour développer un algorithme d'optimisation.

Figure 3-9 – Harris Hawks

3.3.2.1 Comportements sociaux et les stratégies de chasse

Le faucon de Harris a un comportement de recherche de nourriture unique car il attaque sa proie avec d'autres membres du groupe tandis que d'autres espèces chassent seules. Ils connaissent les membres de leur famille et essaient d'être conscients de leurs mouvements pendant l'attaque. Les individus du groupe commencent la mission de chasse en prenant leur position sur les perchoirs puis, les uns après les autres font de courts tours puis atterrissent sur des perchoirs assez hauts. De cette manière, les faucons effectueront occasionnellement un mouvement de « saute-mouton » sur tout le site cible. Ils se rejoindront et se sépareront plusieurs fois pour rechercher l'animal couvert, qui est généralement un lapin.

Les faucons de Harris peuvent démontrer une variété de stratégie de chasse en fonction de la nature dynamique des circonstances et des schémas d'évasion d'une proie, mais leur principale tactique pour capturer une proie est le « bond surprise », également connu sous le nom de stratégie des « sept victimes », qui a été expliquée précédemment.

Une tactique d'échange se produit lorsque le meilleur faucon (Leader) s'arrête devant la proie et se perd, les autres membres continuent la poursuite à la fin, Le principal avantage de ces tactiques coopératives est que les faucons de Harris peuvent poursuivre le lapin détecté jusqu'à l'épuisement, ce qui augmente sa vulnérabilité. De plus, en laissant perplexe la proie

qui s'échappe, il ne peut pas récupérer ses capacités défensives et enfin, il ne peut pas échapper au siège de l'équipe affrontée puisque le faucon le plus puissant et le plus expérimenté capture le lapin fatigué et le partage avec les autres membres du groupe.

Figure 3-10 – Changement de la stratégie d'attaque

3.3.2.2 Optimisation des faucons de Harris (HHO)

La figure suivante montre toutes les phases du HHO (Exploration et Exploitation) :

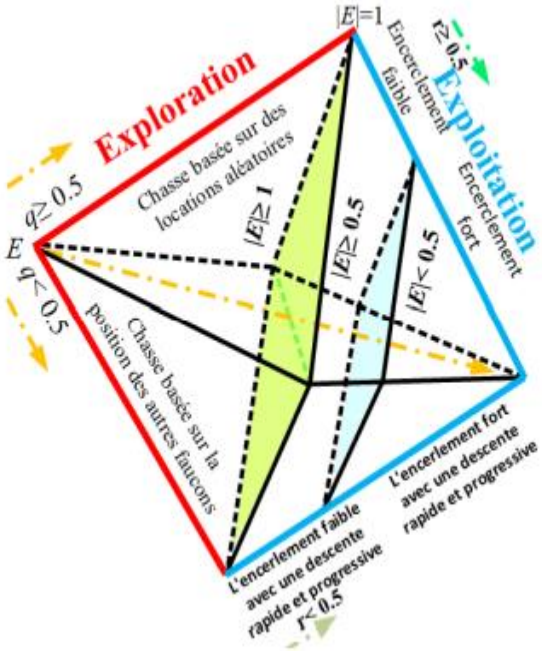

Figure 3-11 – Les phases de l'algorithme HHO [50]

- **Phase d'exploration**

Si nous considérons la nature des faucons de Harris, ils ont des yeux aiguisés qui peuvent les aider à surveiller et à découvrir la proie, mais parfois la proie ne peut pas se voir facilement. Donc, les faucons attendent, observent et surveillent le site désertique pour détecter une proie peut-être après plusieurs heures.

Dans ce mécanisme d'exploration de HHO, les faucons de Harris sont les solutions candidates, alors que la proie est considérée comme le meilleur candidat solution dans chaque itération.

Les faucons de Harris se perchent au hasard sur certains emplacements et attendre pour détecter et attaquer une proie selon deux stratégies représentées par l'équation suivante :

$$\mathbf{X_{rand}^{t+1}} = \begin{cases} \mathbf{X_{rand}^{t} - r1|X_{rand}^{t} - 2r_2\, X^{t}} & \mathbf{si\ q \geq 0.5} \\ \mathbf{(X_{rabbit}^{t} - X_{m}^{t}) - r_3(LB + r_4(UB - LB))} & \mathbf{si\ q \leq 0.5} \end{cases} \qquad (3\text{-}18)$$

- La première règle de l'équation, représente la génération aléatoire de solutions ;
- La deuxième règle de l'équation, représente la différence entre la position de la meilleure solution (lapin) et la position moyenne du groupe.

Tel que :

r3 : Un coefficient aléatoire destiné à accroître la diversité de la recherche

r1, r2, r3, r4 et q : Des nombres aléatoires en (0,1)

La position moyenne des faucons peut être définie comme indiqué dans l'équation suivante :

$$\mathbf{X_i^t} = \frac{1}{N}\sum_{I=1}^{N} \mathbf{X_i^t} \qquad (3\text{-}19)$$

X_i^t représente la position de chaque solution (faucon) dans l'itération t, N indique le nombre total de solutions (faucons)

- **Balance entre diversification (exploration) et intensification (exploitation)**

L'algorithme HHO peut transférer de l'exploration à l'exploitation en fonction de l'énergie d'échappement E de la proie. Mathématiquement cette énergie est modélisée comme :

$$\mathbf{E = 2E_0\left(1 - \frac{t}{T}\right)} \qquad (3\text{-}20)$$

Dont T est le nombre d'itérations maximum, E0 est l'énergie initiale de la proie et il change aléatoirement dans l'intervalle (-1,1) à chaque itération. N est le nombre total de solutions (faucon).

Le statut de la proie est indiqué comme suit :

$$\textbf{Prey status} = \begin{cases} \textbf{Prey is very weak} & \textbf{if } -1 \leq E_0 \leq 0 \\ \textbf{Prey is powerful} & \textbf{if } 0 < E_0 \leq 1 \end{cases} \quad (3\text{-}21)$$

L'algorithme HHO bascule entre diversification (exploration) et intensification (exploitation) en modifiant la stratégie de recherche basée sur l'énergie E de la proie comme suit :

$$\textbf{Stratégie de recherche} = \begin{cases} \textbf{Phase d'exploration Si } |E| \geq 1 \\ \textbf{Phase d'exploitation Si } |E| < 1 \end{cases} \quad (3\text{-}22)$$

Dans HHO, E_0 change aléatoirement à l'intérieur de l'intervalle (- 1, 1) à chaque itération. Lorsque la valeur d'E_0 diminue de 0 à -1, le lapin est physiquement en train de faiblir, tandis que lorsque la valeur d'E_0 augmente de 0 à 1, cela signifie que le lapin se renforce.

La dynamique d'échappement de l'énergie E a une tendance décroissante au cours des itérations.

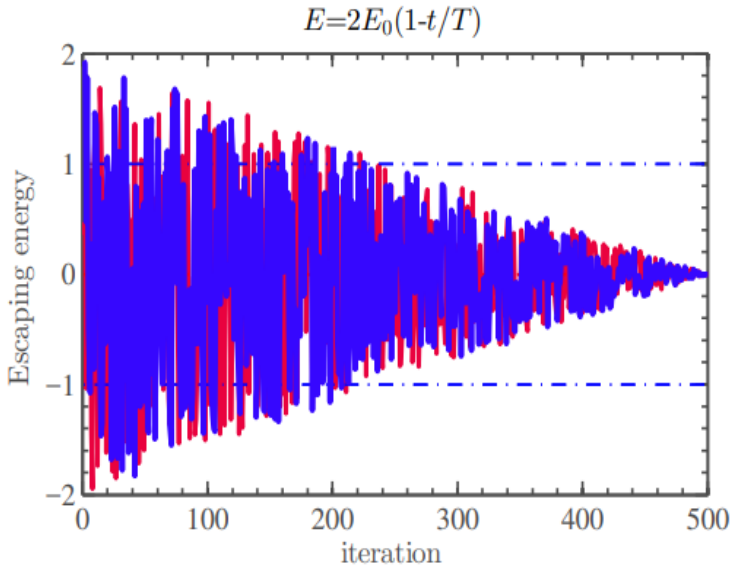

Figure 3-12 – Comportement de E pendant deux passages et 500 itérations [50]

- **Phase d'exploitation**

Dans cette phase, les faucons de Harris effectuent le pounce surpris (sept victimes) en attaquant la proie prévue détectée dans la phase précédente. Cependant, la proie tente souvent d'échapper aux situations menaçantes. Par conséquent, différents styles de chasse se produisent dans des situations réelles.

- **L'encerclement faible**

Si la proie a peu énergie, elle essaie d'échapper aux faucons en faisant des sauts aléatoires, ce moment, les faucons de Harris entourent doucement la proie pour l'épuiser et ensuite l'attaquer d'une façon surprise.

On peut les représenter comme suit :

$$\boldsymbol{X_{t+1}} = \Delta \boldsymbol{X}^{\,t} - \mathbf{E}|\mathbf{J}\mathbf{X}^{t}_{rabbit} - \mathbf{X}^{\,t}| \qquad (3\text{-}23)$$

$$\Delta \boldsymbol{X}^{\,t} = \mathbf{X}^{\,t}_{rabbit} - \mathbf{X}^{t} \qquad (3\text{-}24)$$

Tel que :

$\Delta X^{\,t}$ est la différence entre le vecteur de position du lapin et l'emplacement actuel dans l'itération t, J = 2 (1-r5) représente la force de saut aléatoire du lapin tout au long de la procédure de fuite. Vu que r5 est un nombre aléatoire à l'intérieur de (0,1), et la valeur J change aléatoirement à chaque itération pour simuler la nature des mouvements de lapin (proie).

- **L'encerclement fort**

Dans cette situation, les positions actuelles sont mises à jour à l'aide de l'Equation suivante :

$$\mathbf{X}^{t+1} = \mathbf{X}^{\,t}_{rabbit} - \mathbf{E}|\Delta \mathbf{X}^{t}| \qquad (3\text{-}25)$$

- **L'encerclement faible avec une descente rapide et progressive**

Les faucons de Harris peuvent effectuer l'encerclement faible en décidant de leur prochaine position et en ajustant leur mouvement en comparant le résultat du saut actuel et précédent :

$$\mathbf{Y} = \mathbf{X}^{t}_{rabbit} - \mathbf{E}|\mathbf{J}\mathbf{X}^{t}_{rabbit} - \mathbf{X}^{t}| \qquad (3\text{-}26)$$

Si ($|E| \geq 0{,}5$), la proie peut réussir à s'échapper $r < 0{,}5$. Donc, les faucons de Harris appliquent un encerclement faible (soft) pour l'attaquer.

On peut simuler le mouvement en zigzag de la proie pendant le processus d'évasion à l'aide d'un opérateur de Levy flight (LF).

$$\mathbf{Z} = \mathbf{Y} + \mathbf{S} \times \mathbf{LF}(\mathbf{D}) \qquad (3\text{-}27)$$

Tel que :

D : La dimension du problème.

S : Un vecteur aléatoire de taille 1XD.

LF : calculé comme suite :

$$\mathbf{LF}(\mathbf{X}) = \mathbf{0.01} \times \frac{\mu \times \sigma}{|v|^{\frac{1}{\beta}}}, \sigma = \left(\frac{\tau(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\tau\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right)^{\frac{1}{\beta}} \qquad (3\text{-}28)$$

Où u, v sont des valeurs aléatoires à l'intérieur de (0,1), β est une constante par défaut fixée à 1,5.

On peut effectuer la stratégie finale de mettre à jour les positions des faucons par :

$$X^{t+1} = \begin{cases} \mathbf{Y} & \text{si } \mathbf{f}(\mathbf{Y}) < \mathbf{f}(\mathbf{X}(\mathbf{t})) \\ \mathbf{Z} & \text{si } \mathbf{f}(\mathbf{Z}) < \mathbf{f}(\mathbf{X}(\mathbf{t})) \end{cases} \qquad (3\text{-}29)$$

Où Y et Z sont calculées par les équations (3. 26) et (3. 27).

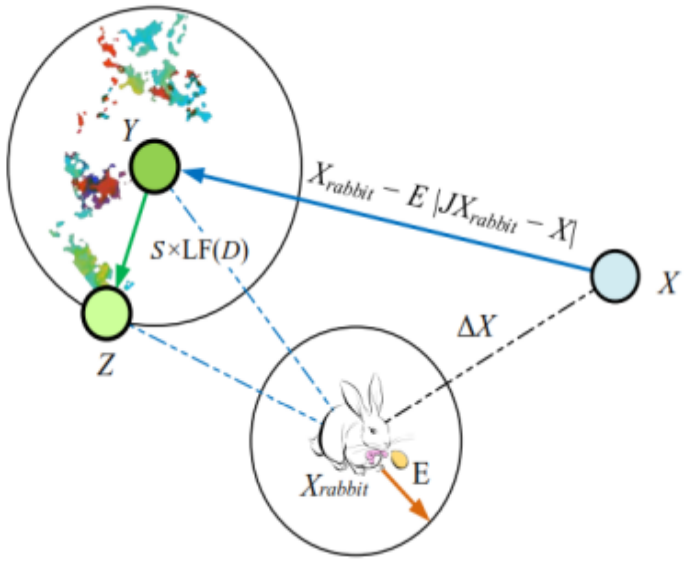

Figure 3-13 – Exemple de vecteurs globaux dans le cas d'encerclement faible avec descentes rapides Progressives [50]

- **L'encerclement fort et une descente progressive rapide**

Lorsque la proie a une énergie faible pour s'échapper (|E| < 0,5) et qu'elle a une chance de s'échapper avec succès r < 0,5, les faucons de Harris appliquent la stratégie d'encerclement fort. Et afin de réaliser cette stratégie, ils tentent de réduire la distance de leur position moyenne X_m avec la proie.

Ce processus est présenté comme suit :

$$\boldsymbol{X^{t+1}} = \begin{cases} \mathbf{Y} & \mathbf{si\ f(Y)\ <\ f(X(t))} \\ \mathbf{Z} & \mathbf{si\ f(Z)\ <\ f(X(t))} \end{cases} \qquad (3\text{-}30)$$

Où Y et Z sont calculées par les équations (3. 26) et (3. 27).

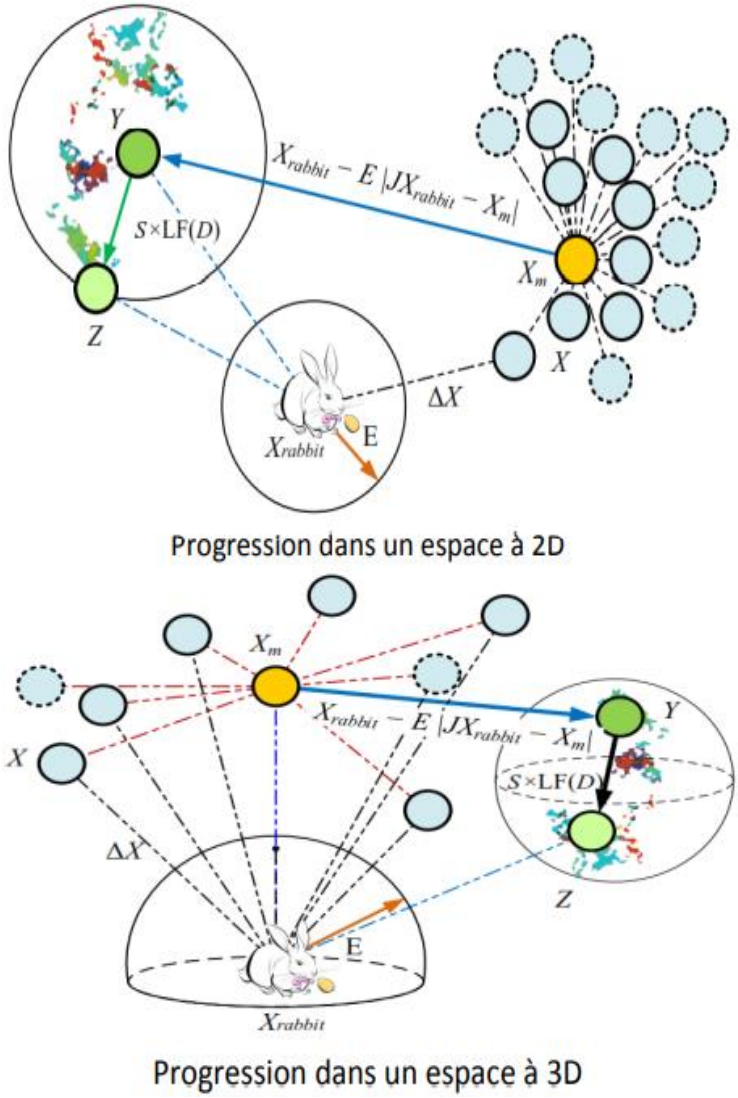

Figure 3-14 – Le processus d'encerclement fort avec une descente progressive rapide dans un espace à 2D/3D [50]

L'organigramme de l'algorithme HHO est représenté dans la figure suivante :

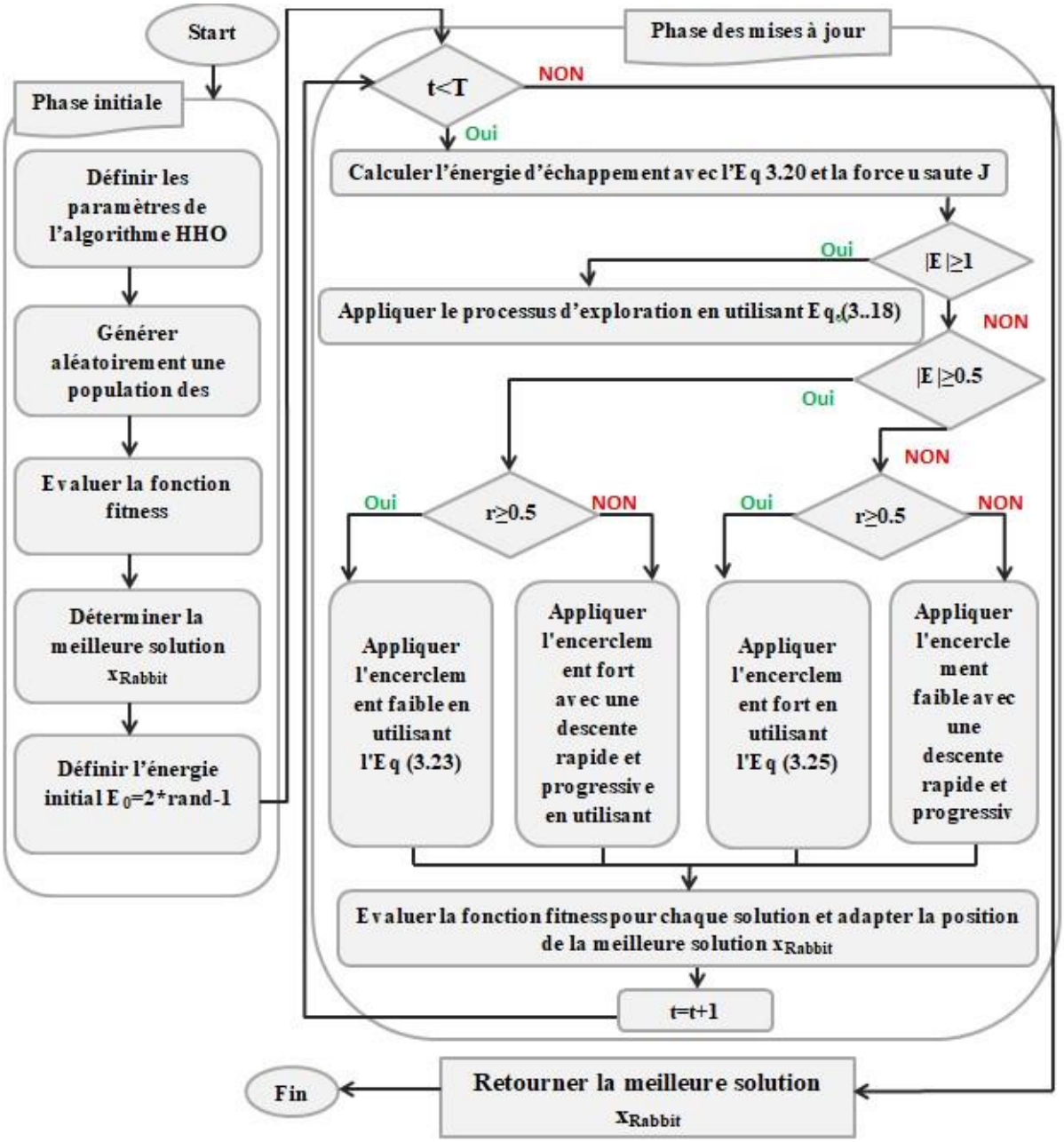

Figure 3-15: Organigramme de l'algorithme HHO

3.4 Le concept de la sélection d'attributs par les approches bio-inspirées

Afin d'appliquer méthodes d'optimisation pour la sélection d'attributs, il faut un codage adéquat de la solution et un choix judicieux de la fonction fitness [18].

Structure de l'individu

Alors, pour sélectionner les termes les plus importants dans un document, il faut générer aléatoirement des valeurs entre 0 et 1, si la valeur dépasse 0,5 alors le terme sera sélectionné sinon il sera éliminé.

La fonction fitness

Afin de quantifier les solutions, on doit mesurer la fitness qui représente un compromis entre l'erreur et le taux de la sélection d'attributs en utilisant la formule suivante :

$$\mathbf{fitness} = \boldsymbol{\alpha} * (\mathbf{1} - \mathbf{Acc}) + \boldsymbol{\beta} * \frac{\boldsymbol{Attributs\ sélectionnées}}{\boldsymbol{D}} \quad (3\text{-}31)$$

$$\boldsymbol{\beta} = \mathbf{1} - \boldsymbol{\alpha} \ \& \ \boldsymbol{\alpha} = \mathbf{0.99} \quad (3\text{-}32)$$

3.5 Conclusion

On assiste ces dernières années à un retour à la nature, en s'inspirant de ses différents phénomènes, et la recherche dans le domaine des systèmes bio-inspirés connait une grande progression, permettant des solutions alternatives de plus en plus performantes dans tous les domaines.

Dans ce chapitre, nous présentons d'abord les différentes inspirations utilisées de manière générale. Ensuite, nous présentons l'algorithme Génétique en détaillant les différentes composantes de l'algorithme Génétique et la progression du processus d'optimisation. Sur la base de la littérature, nous soulignons que les GAs nécessitent un temps d'exécution important car l'encodage utilisé est de type binaire et il existe un risque de convergence vers des minima locaux. Cette faille majeure a donné naissance à une nouvelle famille inspirée de l'intelligence distribuée, qui a été expliquée en détail, comprenant deux algorithmes, l'algorithme HHO et l'algorithme GWO. Enfin, nous expliquons le concept de sélection basée sur des méthodes bio-inspirées dans le cadre de la classification d'opinion.

4 Chapitre 4 Résultats et discutions

4.1 Introduction

Dans ce chapitre, nous commencerons par présenter l'architecture globale de notre approche d'analyse de sentiments. Ensuite nous décrivons le corpus utilisé : AJGT afin de valider notre travail et les outils utilisés. Après, nous montrons les résultats de la phase de prétraitement incluant le nettoyage des données, normalisation, tokenization, Suppression des mots vides, et la suppression des mots. Ensuite nous décrivons les différentes métriques utilisées. Enfin nous montrons les différents résultats de l'identification des opinions à base de k-NN et les approches bio-inspirés GA, HHO, GWO.

4.2 L'architecture du système

Avant de détails les résultats obtenus nous représentants dans la figure suivante l'architecture globale de notre approche d'analyse de sentiments.

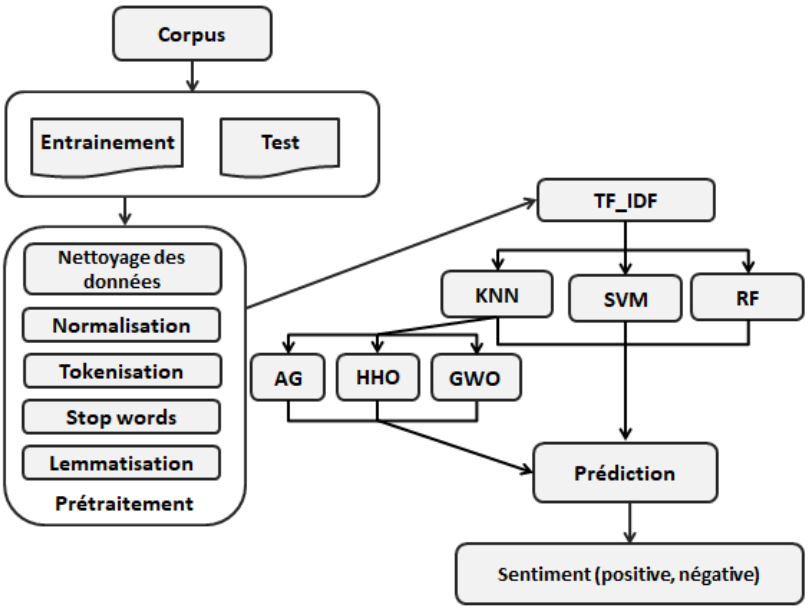

Figure 4-1 – Architecture du système

Pour réaliser notre system nous avons appliqué les prétraitements aux corpus AJGT ensuite la technique TF-IDF pour extraire les feautures à partir des tweets du corpus puis

l'application des trois algorithmes différents Support Vector Machines (SVM), K-plus proche voisin (KNN) et Radom Forest (RF).

4.3 Corpus utilisé

Nous avons effectué une évaluation sur la base de données AJGT qui est un Corpus accessible au public, constitué de 1800tweets (dialecte jordanienne MSA) et qui vont être classifiés par des experts humains selon leur positivité et leur négativité (900 positifs, 900 négatifs). Utilisé dans le but de traiter la classification des opinions afin d'évaluer la performance des modèles présentés dans les chapitres précédents.

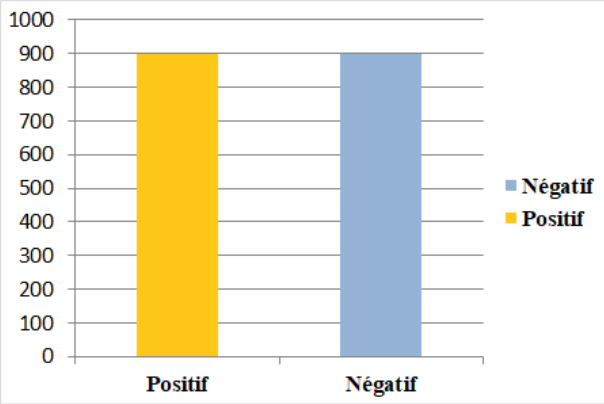

Figure 4-2 – Nombre de tweets

Le tableau suivant montre plus de détail portant sur le corpus utilisé pour tester les approches proposées :

Tableau 4-1 – Détail de base de données traitée

Corpus	Nombre de topics	Nombre de documents	Taille	Type
AJGT	2	1800	1,7 ko	CSV Microsoft Excel.

Pour l'emploie de chaque algorithme il est nécessaire d'utiliser une combinaison de deux ensembles diviser à partir du corpus le premier qui représente un ensemble qu'à partir de lui le système va extraire les connaissances appelé l'ensemble d'apprentissage (training sets), le deuxième est composé des données à classifier appelée l'ensemble de test (test sets).

La division se fait comme suivante : 80% des données sont des trainings sets et 20% test sets comme montre le graphe suivant :

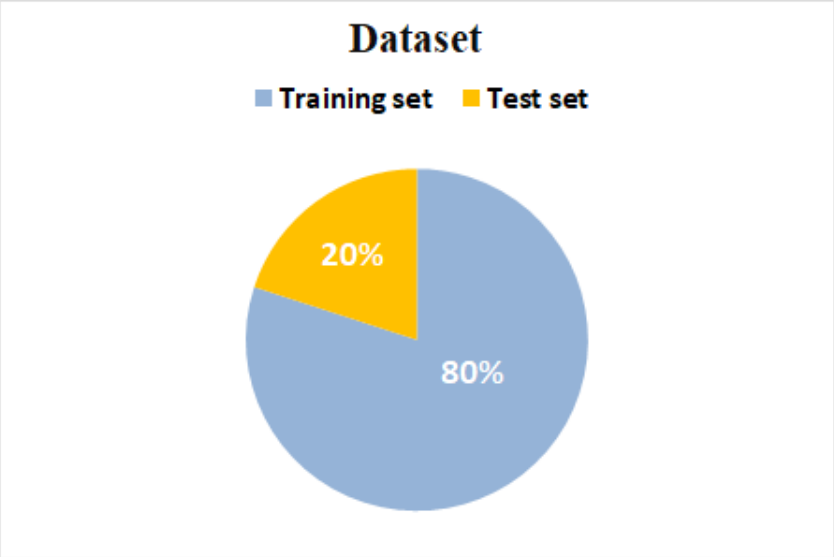

Figure 4-3 – Division des données

4.4 Les outils et librairie utilisé

- **Python**

Figure 4-4 – Logo du Python

Est un langage de programmation open source interprété, multi-paradigme Et multiplateformes de haut niveau créé par Guido van Rossum et sorti en 1991. Python a un système de typage portable, dynamique, extensible, libre et syntaxiquement simple, code plus court que C ou Java, multithread, orienté objet, extensible ; Également, nous avons utilisé le package CSV (Une bibliothèque grâce à lui, nous pouvons manipuler les fichiers de format csv).

- **Google Colabe**

Figure 4-5 – Logo du Google Colabe

Google Colaboratory, souvent raccourci en "Colab" est un service cloud, développé par Google, destiné à la formation et à la recherche dans l'apprentissage automatique ou d'apprentissage en profondeur. À l'exception d'un navigateur on n'a pas besoin d'installer quoi que ce soit sur notre ordinateur pour entraîner des modèles de Machine Learning directement dans le cloud, cet environnement nous permet d'écrire et d'exécuter du code Python dans votre navigateur, avec aucune configuration requise, accès gratuit aux GPU, partage facile.

Google Colab peut être défini comme une version améliorée de Jupyter Notebook. Cette plateforme contient plusieurs bibliothèques très utiles dans le prétraitement linguistique comme le NLTK (Natural language toolkit) et Scikit-learn (une bibliothèque libre Python destinée à l'apprentissage automatique) qui a boosté les résultats de la classification.

- **Pandas**

Figure 4-6– Logo du Pandas

Est l'une des bibliothèques Python les plus utilisées pour la Data Science. Permet la manipulation et l'analyse des données. Elle propose en particulier des structures de données et des opérations de manipulation de tableaux numériques et de séries temporelles.

- **Scikit-learn :**

Figure 4-7– Logo du Scikit-learn

Scikit-learn est une bibliothèque d'apprentissage automatique en Python, le moteur qui alimente de nombreuses applications d'intelligence artificielle et d'exploration de données. Elle concentrer sur les outils d'apprentissage automatique, y compris les algorithmes mathématiques et statistiques qui constituent la base de nombreuses techniques d'apprentissage automatique.

4.5 Résultat de prétraitement

Pour entraîner notre modèle, nous devons d'abord traiter notre ensemble de données en suivant ces étapes : Nettoyage des données, Normalisation, Tokenization, Suppression des mots vides, stemming.

Dans le tableau suivant, nous avons un exemple sélectionné au hasard dans la base de données pour laquelle différentes étapes de prétraitement ont été appliquées :

Tableau 4-2 – Exemple de résultats de l'analyse de prétraitement.

Phrase	الاردن كان افضل حالا بمئات المرات في التعليم و الصحه و الزراعه و الاستثمار ... التردي ليس مرتبط بالربيع العربي بل قبله بمده عام
Nettoyage des données	الاردن كان افضل حالا بمئات المرات في التعليم و الصحه و الزراعه و الاستثمار التردي ليس مرتبط بالربيع العربي بل قبله بمده عام
Normalisation	الاردن كان افضل حالا بمءات المرات في التعليم و الصحه و الزراعه و الاستثمار التردي ليس مرتبط بالربيع العربي بل قبله بمده عام
Tokenization	الاردن,كان,افضل,حالا,بمءات,المرات,في,التعليم,و,الصحه,و,الزراعه,و,الاستثمار,التر دي,ليس,مرتبط,بالربيع,العربي,بل,قبله,بمده,عام
Suppression des mots vides	الاردن,افضل,حالا,بمءات,المرات,التعليم,الصحه,الزراعه,الاستثمار,التردي,مرتبط,بال ربيع,العربي,قبله,بمده,عام
Stemming	ارد فضل حلا بمء رات علم صحه زرع ثمر ترد ربط ربع عرب قبل بمد عام

4.6 Extraction des features

4.6.1 TFIDF (Term Frequency and Inverse Document Frequency)

TF-IDF est une approche efficace utilisée dans de nombreux outils de quering car elle attribue une importance aux mots qui apparaissent rarement mais pas trop, tout en limitant l'importance des mots qui apparaissent fréquemment en utilisant une matrice de nombres pour représenter les mots dans un document.

$$\mathbf{tf(t) = n(t)} \qquad (4\text{-}1)$$

Nous calculons la fréquence du terme dans un document, n(t) étant le nombre de fois où le terme apparaît dans le document. Plus TF est élevé, plus le mot a de l'importance.

$$\mathbf{IDF_t = \log\left(\frac{N}{n_t}\right)} \qquad (4\text{-}2)$$

N représente le nombre de documents et nt le indique nombre de documents où le terme est présent et qui doit être compris 7documents et 1440(80% des documents). Pour un document en particulier on a :

$$\boldsymbol{TFidf(t) = tf(t) \times idf(t)} \qquad (4\text{-}3)$$

Cette étape consiste à transformer l'ensemble de données en une matrice composée de tous les mots de chaque corpus avec leur fréquence d'occurrence, en créant des attributs pour effectuer la tâche de classification basée sur la sélection d'attributs.

4.7 Les paramètres des algorithmes :

Dans cette étape, trois techniques ont été exploitées et comparées: basée sur une méthode inspirée du génétique GA, inspirée du comportement des faucons, et une autre inspirée du comportement des loups gris GWO. Ils ont été testés plusieurs fois pour éliminer l'effet de la génération aléatoire mais le nombre maximal était de 100 itérations ensuite l'utilisation de 60 solutions de recherche (N=60).

4.8 Résultats et discussion

4.8.1 Les mesures d'évaluations

Pour pouvoir valider le modèle proposé, certaines métriques doivent être évaluées selon la matrice de confusion. Quatre mesures sont utilisées, telles que l'accuracy, la précision, le taux de rappel et le F-score.

Tableau 4-3: matrice de confusion

Actual	Predicted		
	1	0
1	TP	FN
0	FP	TN

- TN : le classifieur arrive à identifier le document comme étant un avis négatif.
- TP : Le classifieur arrive à identifier le document comme étant un avis positif.
- FN : le classifieur identifie l'opinion comme étant un avis négatif sachant que l'étiquetage réel indique que l'avis est positif.
- FP : le classifieur identifie l'opinion comme étant un avis positif sachant que l'étiquetage réel indique que l'avis est négatif.

- **Accuracy (Taux de reconnaissance)**

Cette mesure signifie le pourcentage des opinions qui sont correctement classifiés, définie par l'équation suivante :

$$\textbf{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \qquad (4\text{-}4)$$

- **Précision**

La précision est la proportion des avis positifs au niveau de la prédiction obtenue par le classifieur.

$$\textbf{Précision} = \frac{\textbf{TP}}{\textbf{TP+FP}} \qquad (4\text{-}5)$$

- **F mesure**

Une mesure qui est exprimée en fonction de la précision et le rappel c'est à dire la moyenne harmonique, nommée F-mesure ou F-score :

$$\textbf{F score} = 2 \times \frac{(\textbf{Précision} \times \textbf{Rappel})}{(\textbf{Précision} + \textbf{Rappel})} \qquad (4\text{-}6)$$

- **Sensibilité (Taux de rappel, Recall en anglais)**

Cette mesure permet de mesurer la proportion des avis positifs réellement qui sont correctement identifiés, et écrite sous la forme suivante :

$$\textbf{Rappel} = \frac{\textbf{TP}}{\textbf{TP+FN}} \qquad (4\text{-}7)$$

4.8.2 Classification

À base de RF, SVM, K-NN

Le [Tableau 4-4] montre l'évaluation de la métrique d'accuracy en utilisant la base de données AJGT, en appliquant les classificateurs RF, SVM, K-NN.

D'après les résultats, nous remarquons que la performance de K-NN en termes d'accuracy sur ce corpus évaluant les opinions positifs et négatifs est de 68%, le résultat obtenu est diminuée par rapport aux RF et SVM qui ont montré leur supériorité par un taux de 81%, 82% consécutivement.

Tableau 4-4 – Les résultats d'accuracy par RF, SVM, K-NN

Fraction du data set	RF	SVM	KNN
80% | 20%	81%	82%	68%

4.8.3 Classification des opinions à base de la sélection d'attributs :

L'application de la classification à base de la sélection d'attributs en utilisant GA HHO ou GWO avec le KNN a généré plusieurs résultats, le but n'est pas d'avoir un meilleur taux seulement mais en même temps de minimiser le nombre d'attributs utiliser pour cella nous avons utilisé la fonction de minimisation et maximisation de taux d'accuracy.

Les résultats obtenus sont les suivants :

- **L'algorithme GA :**
  - Avec la fonction fitness maximisation : le taux d'accuracy est 89.11% avec un nombre de paramètres de 232 attributs et best fitness de 0.89.

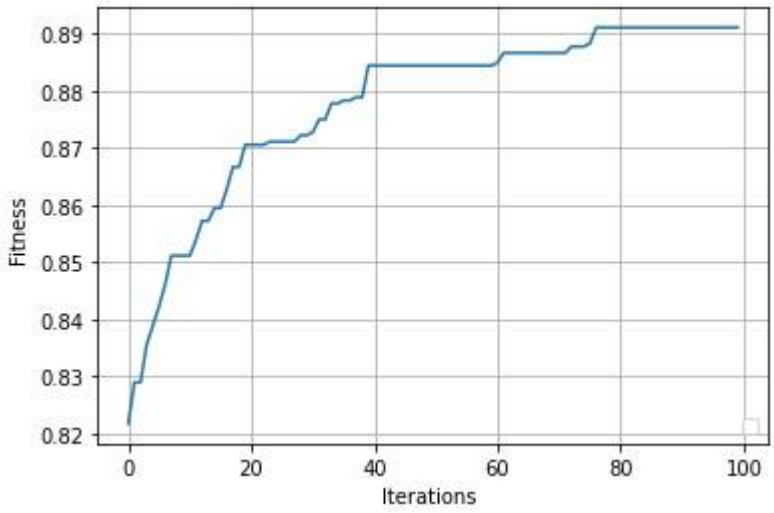

Figure 4-8 – Fitness en fonction des itérations en utilisant la méthode GA avec la fonction fitness maximisation

Selon la figure, nous annotons que la valeur du fitness est en augmentation en termes d'itérations.

- Avec la fonction de fitness de minimisation : le taux d'accuracy est 89.33% avec un nombre de paramètres de 226 attributs et best fitness de 0.11

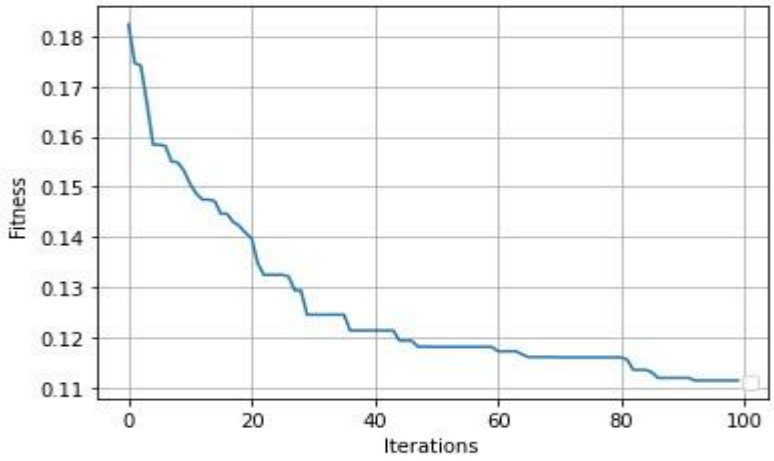

Figure 4-9 – Fitness en fonction des itérations en utilisant la méthode GA avec la fonction fitness minimisation

D'après la figure nous remarquons la diminution de la valeur fitness avec l'augmentation de nombre d'itérations.

- **L'algorithme HHO :**
  - Avec la fonction fitness maximisation : le taux d'accuracy est 82% avec un nombre de paramètres de 254 attributs avec une best fitness de 0.827

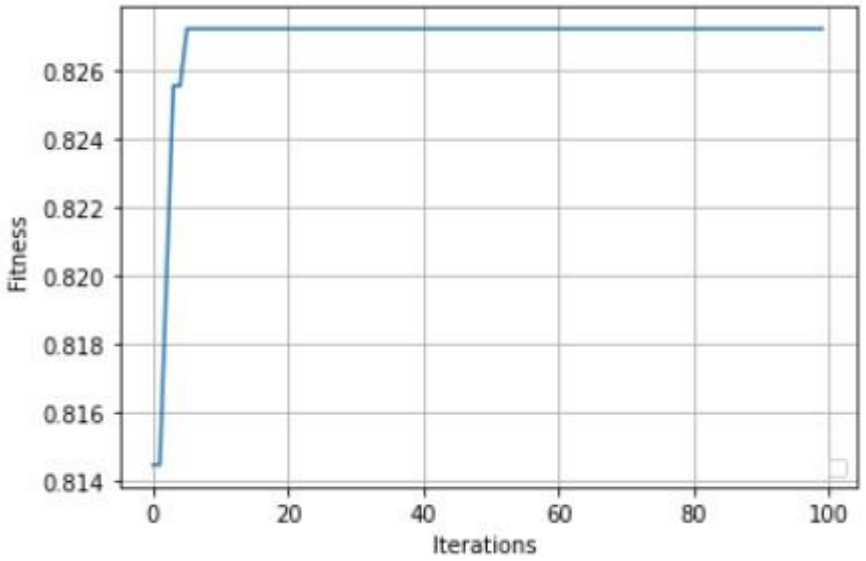

Figure 4-10 – Fitness en fonction des itérations en utilisant la méthode HHO avec la fonction fitness maximisation

Dans ce cas nous remarquons que la valeur de la fitness a été augmentée après seulement quelques itérations et fixé à la valeur 0.827

  - Avec la fonction de fitness de minimisation : le taux d'accuracy est 83% avec un nombre de paramètres de 300 attributs avec une best fitness de 0.17

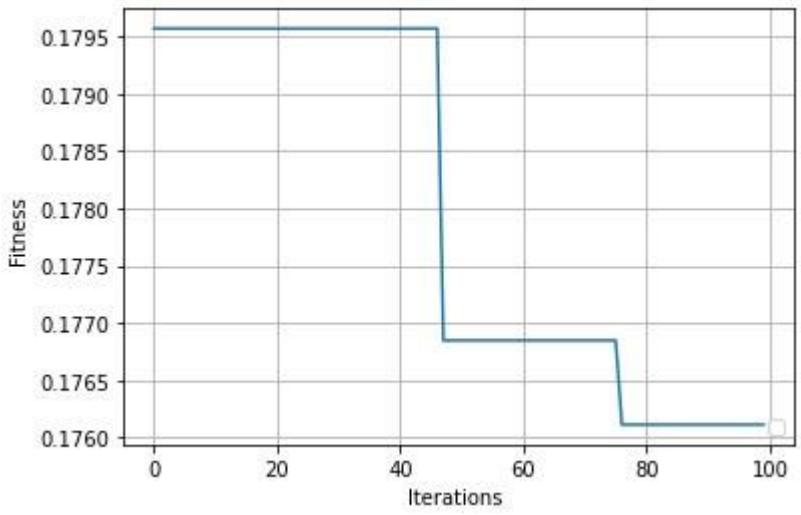

Figure 4-11 – Fitness en fonction des itérations en utilisant la méthode HHO avec la fonction fitness minimisation

- **L'algorithme GWO :**
  - Avec la fonction fitness maximisation : le taux d'accuracy est 85.27% avec un nombre de paramètres de 312 attributs avec une best fitness de 0.85

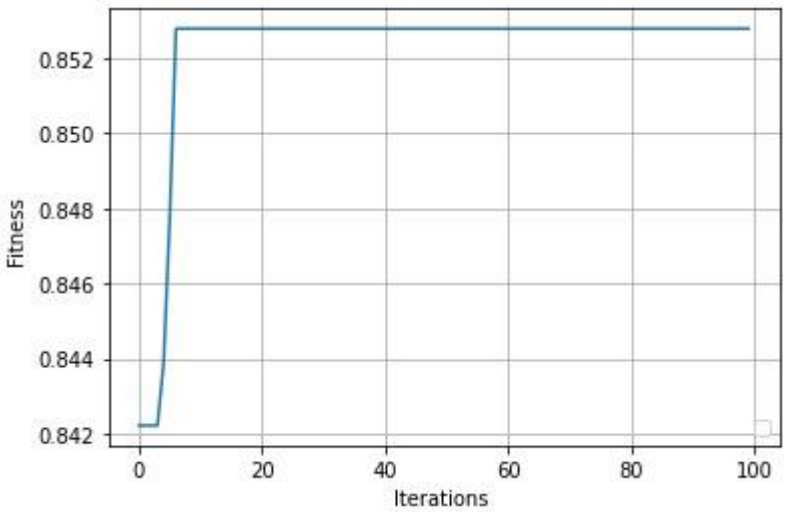

Figure 4-12 – Fitness en fonction des itérations en utilisant la méthode GWO avec la fonction fitness maximisation

De la figure, nous constatons que la valeur du fitness a augmenté directement à 0,85, puis elle est devenue stable.

- Avec la fonction de fitness de maximisation : le taux d'accuracy est 85% avec un nombre de paramètres de 325attributs avec une best fitness de 0.15

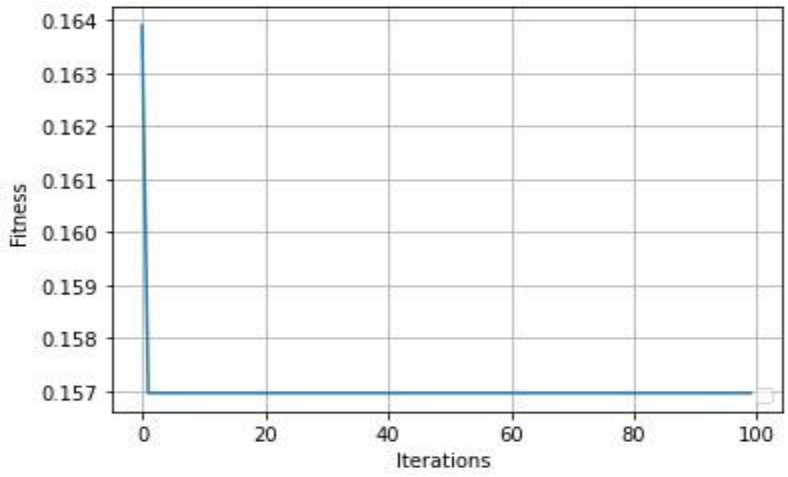

Figure 4-13 – Fitness en fonction des itérations en utilisant la méthode GWO avec la fonction fitness minimisation

La figure nous permis de constater que la valeur de fitness a été directement diminuer à 0.15 et a gardé sa stabilité malgré le nombre d'itérations essayer.

L'application de GA sur le KNN nous a permis d'augmenter le taux de reconnaissance avec une marge de 21% par rapport au classifieur k-NN. Pour K-NN-HHO, les résultats obtenus montrent une amélioration considérée au niveau de la reconnaissance avec une marge de 15%, tandis que K-NN-GWO atteint une performance avec un taux de 85%.

L'histogramme représente la différence de performance de K-NN en termes d'accuracy avec les trois algorithmes :

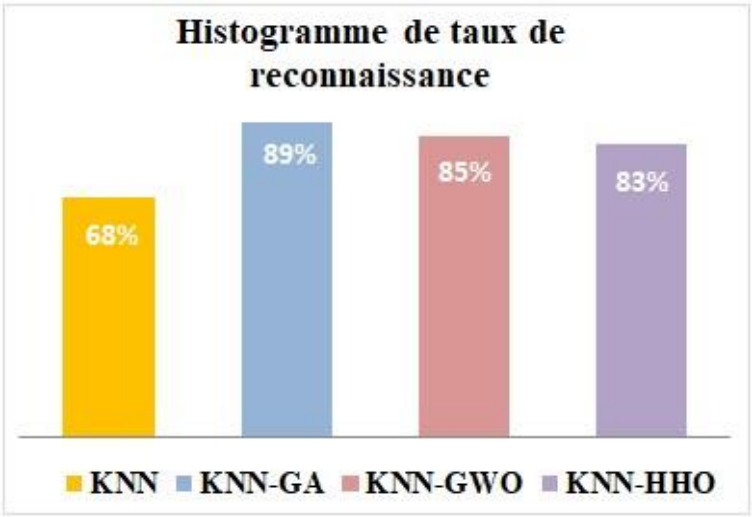

Figure 4-14 – Histogramme de taux de reconnaissance du KNN et KNN avec GA, HHO et GWO

4.8.4 Evaluation des métriques en termes

- **D'Accuracy (Taux de reconnaissance)**

Le [tableau 4-5] montre l'évaluation de la métrique d'accuracy en utilisant la base de données AJGT, en appliquant l'approche GWO, HHO et GA pour réaliser la sélection d'attributs à base de classifieur K-NN Nous constatons que le taux de reconnaissance dépasse 82%. Nous observons que l'algorithme GA donne un pourcentage d'accuracy très élevé par rapport à GWO et HHO avec un taux de 89%.

Tableau 4-5 – Résultats de la métrique d'accuracy obtenus

La fonction fitness de maximisation			**La fonction fitness de minimisation**		
GA	**HHO**	**GWO**	**GA**	**HHO**	**GWO**
89%	82%	85%	**89%**	83%	85%

- **Sensibilité (Recall)**

[Le tableau 4-6] montre l'évaluation de la métrique Sensibilité en utilisant GWO/HHO/GA sur AJGT. Nous remarquons clairement que le meilleur taux de rappel moyen est obtenu par GA avec les deux fonctions: 89%.

Tableau 4-6 – Résultats de la métrique sensibilité obtenus

La fonction fitness de maximisation			La fonction fitness de minimisation		
GA	**HHO**	**GWO**	**GA**	**HHO**	**GWO**
89%	83%	85%	**89%**	83%	85%

- **Précision**

La meilleure précision obtenue par l'algorithme GA est de 89% avec les deux fonctions montrées dans le [Le tableau 4-7].

Tableau 4-7 – Résultats de la métrique précision obtenus

La fonction fitness de maximisation			La fonction fitness de minimisation		
GA	**HHO**	**GWO**	**GA**	**HHO**	**GWO**
89%	84%	85%	**89%**	83%	86%

- **F score**

[Le tableau 4-8] montre l'évaluation de la métrique F mesure en utilisant les trois algorithmes. On Remarque que la meilleure valeur de Fscore atteint approximativement 89% avec l'algorithme GA avec les deux fonctions de minimisation et de maximisation.

Tableau 4-8 – Résultats de la métrique Fscore obtenus

La fonction fitness de maximisation			La fonction fitness de minimisation		
GA	**HHO**	**GWO**	**GA**	**HHO**	**GWO**
89%	83%	85%	**89%**	83%	85%

4.9 Conclusion

Dans ce chapitre, nous avons fait l'analyse de sentiments sur un corpus AJGT. Nous avons exploité trois classificateurs d'apprentissage automatique qui sont SVM, RF, KNN. L'évaluation de ces classificateurs se fait par 20% du corpus. Nous avons utilisé ainsi des algorithmes à base de la sélection d'attributs HHO, GWO et GA avec l'algorithme KNN.

Suite aux résultats obtenus après l'exécution de tous les algorithmes cités aux paravents avec les deux fonctions de maximisation et minimisation et la comparaison entre eux, nous avons constaté que l'algorithme GA donne des résultats optimaux que l'algorithme GWO et HHO pour la classification des opinions (positives/ négatives) en traitant notre base de données. Nous pouvons conclure aussi que La méthode GA est plus efficace que les autres méthodes à cause de la génération aléatoire des vecteurs réels suivis par les échanger en des individus binaires, et la focalisation de tous les individus de la population sur la solution optimale.

Conclusion générale

Les réseaux sociaux ont attiré des milliards de personnes pour interagir les unes avec les autres. Aujourd`hui, de nombreuses personnes utilisent quotidiennement ces médias, non seulement comme plate-forme de communication, mais aussi pour partager et échanger des opinions, des commentaires et des expériences. L'importance de l'analyse des sentiments découle de la capacité à tirer et à déduire des conclusions indirectes à partir d'une énorme quantité de données.

L'analyse d'un texte écrit dans une langue à la morphologie complexe, telle que la langue arabe, a toujours été un processus difficile à plusieurs niveaux. C'est pourquoi nous avons d'abord commencé à prétraiter notre ensemble de données en suivant ces étapes : Nettoyage des données, Normalisation, Tokenization, Suppression des mots vides, stemming.

Et vu que les mots exprimant les sentiments, le propriétaire d'opinion, et les informations contextuelles sont les indices dans l'identification des expressions d'opinions et dans la détermination de leurs tendances. Donc, la démarche d'identification est basée sur l'extraction des mots de sentiments en utilisant TF-IDF comme prochaine étape, et ensuite, l'identification des polarités (tendances) des expressions porteuses d'opinions, en premier lieu, par trois algorithmes de classification: SVM, RF, KNN. Ensuite par la sélection d'attributs à partir d'algorithmes bio-inspirés à base de GA, HHO, GWO avec KNN afin d'optimiser la précision. Les résultats sont satisfaisants et montrent que l'utilisation de GA surpasser les autres techniques utilisées avec un pourcentage de 89 en termes d'accuracy, F-score, précision et taux de rappel.

Comme perspectives nous pouvons citer:

- L'intégration d'autres techniques et méthodes de classification ;
- Utilisation d'autre méthode d'optimisation ;
- Augmentation de nombre de données pour minimiser l'erreur ;
- la création d'un nouvel ensemble de donnée arabe collecté auprès de tweeter concernant l'un des sujets tendance tel que "la guerre Russe-Ukraine", puis le traitement de corpus pour connaitre les opinions des arabes sur ce sujet.

Bibliographie

[1] Cambria, E. (2016). Affective computing and sentiment analysis. IEEE, 31(2), 102-107.

[2] Pang, B., & Lee, L. (2009). Opinion mining and sentiment analysis. Comput. Linguist, 35(2), 311-312.

[3] Liu, B. (2020). Sentiment analysis: Mining opinions, sentiments, and emotions. Cambridge university press.

[4] Turney, P. D., & Littman, M. L. (2002). Unsupervised learning of semantic orientation from a hundred-billion-word corpus. arXiv preprint cs/0212012.

[5] Hatzivassiloglou, V., & McKeown, K. (1997 july, july). Predicting the semantic orientation of adjectives. 35th annual meeting of the association for computational linguistics and 8th conference of the european chapter of the association for computational linguistics, (pp. 174-181).

[6] Yu, H., & Hatzivassiloglou, V. (2003). Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. Proceedings of the 2003 conference on Empirical methods in natural language processing, (pp. 129-136).

[7] Al-Kabi, M. N., Abdulla, N. A., & Al-Ayyoub, M. (2013, December). An analytical study of arabic sentiments: Maktoob case study. 8th International Conference for Internet Technology and Secured Transactions (ICITST-2013) (pp. 89-94). IEEE.

[8] Abdulla, N. A., Al-Ayyoub, M., & Al-Kabi, M. N. (2014). An extended analytical study of arabic sentiments. International Journal of Big Data Intelligence, 1(1-2), 103-113.

[9] Omar, N., Albared, M., Al-Moslmi, T., & Al-Shabi, A. (2014, December, December). A comparative study of feature selection and machine learning algorithms for Arabic sentiment classification. Dans C. Springer (Éd.), Asia information retrieval symposium, (pp. 429-443).

[10] Cherif, W., Madani, A., & Kissi, M. (2015). Towards an efficient opinion measurement in Arabic comments. Procedia Computer Science, 73, 122-129.

[11] Duwairi, R. M., Alfaqeh, M., Wardat, M., & Alrabad, A. (2016, April). Sentiment analysis for Arabizi text. Dans IEEE (Éd.), 2016 7th International Conference on Information and Communication Systems (ICICS), (pp. 127-132).

[12] Aliane, A. A., & al, &. (2016). A genetic algorithm feature selection based approach for Arabic sentiment classification. Dans IEEE (Éd.), 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA).

[13] Hadi, W. E. (2015). Classification of Arabic social media data . Advances in Computational Sciences and Technology, 8(1), 29-34.

[14] Alomari, K. M., ElSherif, H. M., & & Shaalan, K. (2017, June). Arabic tweets sentimental analysis using machine learning. Dans Springe (Éd.), International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, (pp. 602-610).

[15] Al-Saqqa, S., & Abdel-Nabi, H. &. (2018). A Survey of Textual Emotion Detection. 2018 8th International Conference on Computer Science and Information Technology (CSIT) (pp. 136-142). doi: 10.1109/CSIT.2018.8486405.

[16] Tubishat, M., Abushariah, M. A., & Idris, N. (2019). Improved whale optimization algorithm for feature selection in Arabic sentiment analysis. Applied Intelligence, 49(5), 1688-1707.

[17] Dhal, K. G., Das, A., Ray, S., Gálvez, J., & Das, S. (2020). Nature-inspired optimization algorithms and their application in multi-thresholding image segmentation. Archives of Computational Methods in Engineering, 27(3), 855-888.

[18] Alzaqebah, A., Smadi, B., & Hammo, B. H. (2020, April). Arabic Sentiment Analysis Based on Salp Swarm Algorithm with S-shaped Transfer Functions. Dans IEEE (Éd.), 2020 11th International Conference on Information and Communication Systems (ICICS), (pp. 179-184).

[19] Boudjnane, A., & Kadri, N. (2019). Classification de texte arabe à l'aide de la méthode d'optimisation Gray Wolf Optimizer. PFE, Université des Sciences et de la Technologie d'Oran Mohamed BOUDIAF.

[20] Montejo-Ráez, A., Martínez-Cámara, E., Martin-Valdivia, M. T., & lopez, L. A. (2012, July). Random walk weighting over sentiwordnet for sentiment polarity detection on twitter. Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and sentiment analysis, (pp. 3-10).

[21] Al-Ayyoub, M., Essa, S. B., & Alsmadi, I. (2015). Lexicon-based sentiment analysis of arabic tweets. International Journal of Social Network Mining, 2(2), 101-114.

[22] Chiavetta, F. B. (2016, April). A Lexicon-based Approach for Sentiment Classification of Amazon Books Reviews in Italian Language. WEBIST, 2, pp. 159-170.

[23] Mertiya, M., & Singh, A. G. (2016). A Novel Approach of Sentiment Detection on Twitter (Doctoral dissertation).

[24] El-Halees, A. M. (2011). Arabic opinion mining using combined classification approach.

[25] Alhumoud , S., Albuhairi, T., & Alohaideb, W. (2015, November). Hybrid sentiment analyser for Arabic tweets using R. 2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), 1, pp. 417-424.

[26] Khalifa, K., & Omar, N. (2014). A hybrid method using lexicon-based approach and Naive Bayes classifier for Arabic opinion question answering. J. Comput. Sci, 10, 10.

[27] Gamallo, P., & Garcia, M. (2014). Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets. Semeval@ coling, 2, pp. 171-175.

[28] KHEMAKHEM, A. (2006). Arabic LDB: une base lexicale normalisée pour la langue arabe. Mémoire de master, Université de Sfax, Tunisie.

[29] BALLAOUI , H. (2017). Le traitement automatique de la langue arabe (TALA) pour la recherche d'information sur le eb. thèse de doctorat, L'Université Chouaïb Doukkali D'El Jadida, Maroc.

[30] Mohammed El Amine, A. (2018). Reconnaissance des unités linguistiques signifiantes. hèse de Doctorat, Université Abou Bekr BELKAID TLEMCEN, Algérie.

[31] Belguith, L. H., & Chaâben, N. (2006, 04). Analyse et désambiguïsation morphologiques de textes arabes non voyellés. (TALN, Éd.)

[32] Belguith, L. H., Baccour, L., & Mourad . (2005). Segmentation de textes arabes basée sur l'analyse contextuelle des signes de ponctuations et de certaines particules. Dans ATALA (Éd.), Actes de la 12ème conférence sur le Traitement Automatique des Langues Naturelles, (pp. 451–456).

[33] Boudad, N., Faizi, R., Oulad Haj Thami, R., & Chiheb, R. (2017). Sentiment analysis in Arabic: A revie of the literature. Ain Shams Engineering Journal.

[34] Attia, M. (2006). An ambiguity-controlled morphological analyzer for modern. Challenges of Arabic for NLP/MT, The British Computer Society. London.

[35] Maraoui, M., Zrigui, M., & Antoniadis, G. (s.d.). Un système de génération automatique de dictionnaires étiquetés de l'arabe. (CITALA, Éd.)

[36] Habash, N., Eskander, R., & Hawwari, A. (2012). A Morphological Analyzer for Egyptian Arabic. NAACL-HLT Workshop on Computational Morphology and Phonology.

[37] Duwair, R. M., Marji, R., Sha'ban, N., & Rushaidat, S. (2014). Sentiment Analysis In Arabic. International conference on Information and Communication Systems (icics).

[38] Oussous, A., Benjellon, F. Z., Lahce, A. A., & Belfkih, S. (2019). ASA: A framework for Arabic sentiment analysis. Journal of Informatic Science.

[39] Alowaidi, S., Saleh, M., & Abulnaja, O. (2017). Semantic Sentiment Analysis of Arabic Texts. International Journal of Advanced Computer Science and Applications, 8.

[40] Taghva, Kazem, & Elkhoury. (2005). Arabic stemming without a root dictionary. Dans IEEE (Éd.), International Conference on Information Technology: Coding and Computing (ITCC'05), 1, pp. 152-157.

[41] Ayodele, T. (2010). Types of Machine Learning Approach. Dans New Advances in Machine Learning (pp. 19-48).

[42] Zrigui, M., Ayadi, R., Mars, M., & Maraoui, M. (2012, 02 20). Arabic Text Classification Framework Based on Latent Dirichlet Allocation. Journal of Computing and Information Technology.

[43] Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. Dans Emerging artificial intelligence applications in computer engineering (pp. 3-24).

[44] Gadhvi, H., & Madhu, S. (2013). Comparative Study of Classification Algorithms forWeb Spam Detection. International Journal of Engineering Research et Technology, 2497-2501.

[45] Tsang, P., & Au, A. (1996). A genetic algorithm for projective invariant object recognition. Dans IEEE (Éd.), Proceedings of Digital Processing Applications (TENCON '96), (pp. 58 – 63).

[46] Storn, R., & Price, K. (1997). Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. Journal of Global Optimization, 11(4), 41–359.

[47] Mirjalili, S., Mirjalili, S., & Lewis, A. (2014). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46-55.

[48] Wang, J., & Li, S. (2019). An Improved Grey Wolf Optimizer Based on Differential Evolution and Elimination Mechanism. Scientific Reports(7181).

[49] Dai, S., Niu, D., & Yan, L. (2018). Daily Peak Load Forecasting Based on Complete Ensemble Empirical Mode Decomposition with Adaptive Noise and Support Vector Machine Optimized by Modified Grey Wolf Optimization Algorithm. Energies, 11(1).

[50] Heidari, A., & al. (2019). Harris hawks optimization: Algorithm and applications. Future Generation Computer Systems, 97, 849-872.