

TP03:

Systemes multi-agents

Janvier, 2024

Imene Bouaziz, ISI G01

Métaheuristiques à base de jeu



Ecole Supérieure en Informatique

Table de matières

- 1- Problématique**
- 2- Métaheuristiques à base de jeu**
- 3- La métaheuristiques basée sur le Ludo (Ludo game-based metaheuristic)**
- 4- Les quatre métaheuristiques à base de jeu choisies**
- 5- Le dataset utilisé**
- 6- Prétaitement**
- 7- Fonction de Fitness**
- 8- Evaluations**

1- Problématique

Le problème des fausses informations, ou "fake news", se caractérise par la dissémination délibérée de contenus inexacts, trompeurs ou fallacieux, souvent présentés comme des faits réels. Amplifié par les médias sociaux et Internet, ce phénomène engendre une propagation rapide de désinformation. Les fausses informations, qu'elles prennent la forme d'articles, d'images ou de vidéos manipulés, peuvent influencer l'opinion publique, miner la confiance dans les médias et perturber le processus démocratique.

Les motivations derrière cette diffusion varient, allant de la désinformation intentionnelle à des intérêts politiques, idéologiques ou économiques. Combattre ce problème nécessite une approche globale incluant l'éducation du public, la vérification des faits, la régulation des plateformes en ligne et une sensibilisation accrue sur la production et la diffusion d'informations.

1- Problématique

Face à la complexité croissante des problèmes contemporains, tels que la recherche d'une solution optimale dans un environnement changeant rapidement, comment la swarm intelligence s'est-elle révélée être une approche pertinente pour aborder ces défis complexes?

Swarm intelligence implique des agents décentralisés interagissant collectivement pour atteindre un objectif commun, s'inspirant du comportement coordonné observé dans la nature afin de résoudre des problèmes complexes. Elle améliore la précision des décisions et la robustesse grâce à la sagesse collective.

2- Métaheuristiques à base de jeu

Les métaheuristiques à base de jeu sont des approches algorithmiques inspirées du comportement stratégique des jeux pour résoudre des problèmes complexes d'optimisation. En émulant la dynamique compétitive des jeux, ces techniques visent à trouver des solutions optimales dans des environnements incertains. Appliquées principalement à des problèmes d'optimisation combinatoire, tel que le problème du voyageur de commerce,

elles exploitent des mécanismes tels que les algorithmes évolutionnaires ou génétiques pour simuler la compétition entre différentes solutions potentielles. L'objectif est de parvenir à des solutions de haute qualité en imitant le processus de sélection naturelle observé dans les jeux, où les solutions les plus performantes survivent et évoluent au fil du temps, offrant ainsi des approches efficaces pour résoudre des problèmes d'optimisation complexes.

3- La métaheuristiques basée sur le Ludo (Ludo game-based metaheuristic)

La métaheuristique du jeu de ludo, également connue sous le nom de "Ludo Game Metaheuristic," est une approche algorithmique qui s'inspire du jeu de société Ludo pour résoudre des problèmes d'optimisation. En utilisant les règles et la dynamique du jeu Ludo comme base, cette métaheuristique modélise le processus de recherche de solutions optimales à travers l'émulation du déplacement stratégique des pions sur le plateau. Les mécanismes de mouvement, les choix tactiques et la compétition entre les pions dans le jeu sont adaptés pour représenter différentes solutions possibles dans l'espace de recherche d'un problème d'optimisation spécifique.

En appliquant cette métaheuristique, on cherche à bénéficier de la nature ludique du jeu Ludo pour concevoir des algorithmes efficaces dans la résolution de problèmes complexes, offrant ainsi une approche originale et créative dans le domaine de l'optimisation algorithmique.



4- Les quatre métaheuristiques à base de jeu choisies

Tug of War optimizer

- L'Optimisation Tug of War (TWO) est un nouvel algorithme métaheuristique basé sur la population, inspiré du jeu de la corde à tirer, développé par Kaveh et Zolghadr.
- L'algorithme utilise une métaphore sportive en considérant chaque solution candidate comme une équipe participant à une série de compétitions de tir à la corde.
- Les équipes exercent des forces de traction les unes sur les autres en fonction de la qualité des solutions qu'elles représentent, et les équipes concurrentes se déplacent vers leurs nouvelles positions selon les lois newtoniennes de la mécanique.
- Les caractéristiques clés de TWO comprennent sa formulation, qui prend en compte les qualités des deux équipes en interaction, le rendant applicable à l'optimisation globale de fonctions discontinues, multimodales, non lisses et non convexes.



4- Les quatre métaheuristiques à base de jeu choisies

Hide Object Game Optimizer

- HOGO (Hide Objects Game Optimization) est un algorithme d'optimisation métaheuristique inspiré du comportement des joueurs dans un jeu de cache-cache. Il modélise le comportement des joueurs cherchant un objet caché dans un espace donné ou le jeu est guidé par la voix de l'entraîneur ; plus elle est forte, plus le joueur est proche de l'objet caché. Les autres joueurs cherchent à rester près du meilleur joueur et loin du pire joueur, et ils s'influencent mutuellement. Il utilise ces informations pour optimiser une fonction objective spécifiée. Le fonctionnement de l'algorithme HOGO peut être résumé ainsi :
- Initialiser un ensemble de solutions candidates de manière aléatoire.
- Évaluer la qualité de chaque solution candidate.
- Sélectionner la meilleure solution en fonction de sa valeur de qualité.
- Générer un nouvel ensemble de solutions candidates en modifiant la meilleure solution en se basant sur le comportement des joueurs dans un jeu de cache-cache.
- Répéter les étapes 2 à 4 jusqu'à ce qu'une condition d'arrêt soit atteinte.



4- Les quatre métaheuristiques à base de jeu choisies

League Championship algorithm

- Le League Championship Algorithm (LCA) est un algorithme d'optimisation mondiale inspiré du sport, simulant le processus de championnat dans les ligues. Introduit par Ali Husseinzadeh Kashan en 2009, il se base sur la compétition entre équipes, où les meilleures montent en divisions supérieures et les moins performantes descendent. Populaire en recherche, le LCA optimise des fonctions continues non linéaires en utilisant la dynamique compétitive des ligues sportives.



4- Les quatre métaheuristiques à base de jeu choisies

Most Valuable Player algorithm

- Le Most Valuable Player Algorithm (MVPA) est un nouvel algorithme d'optimisation inspiré du sport, spécialement du concept du Most Valuable Player (MVP).
- Il résout des problèmes d'optimisation et montre des résultats prometteurs.
- Implementation :
 - Initialisation d'une population de solutions.
 - Évaluation des solutions sur un ensemble de validation.
 - Faire des compétitions individuelles dans la même équipe et entre équipes
 - Identification du "Most Valuable Player" pour chaque itération.
 - Mise à jour de la population par mutation ou croisement pour varier et éviter un optima local.
- **Répétition des étapes 3 et 4 jusqu'à atteindre un critère d'arrêt.**

5- Le dataset utilisé

ISOT Fake News Dataset propose deux types d'articles, réels et faux, collectés respectivement sur Reuters.com et des sources de fausses informations signalées par Politifact et Wikipedia. Les fichiers CSV "True.csv" et "Fake.csv" comprennent plus de **12 600 articles chacun, avec des détails tels que le titre, le texte, le type, et la date de publication.** La collecte s'est principalement concentrée sur les années 2016-2017. Les données, bien que nettoyées, préservent les ponctuations et erreurs des fausses informations. Le tableau ci-dessous détaille les catégories et le nombre d'articles par catégorie.

News	Size (Number of articles)	Subjects	
Real-News	21417	Type	Articles size
		World-News	10145
		Politics-News	11272
Fake-News	23481	Type	Articles size
		Government-News	1570
		Middle-east	778
		US News	783
		left-news	4459
		politics	6841
		News	9050

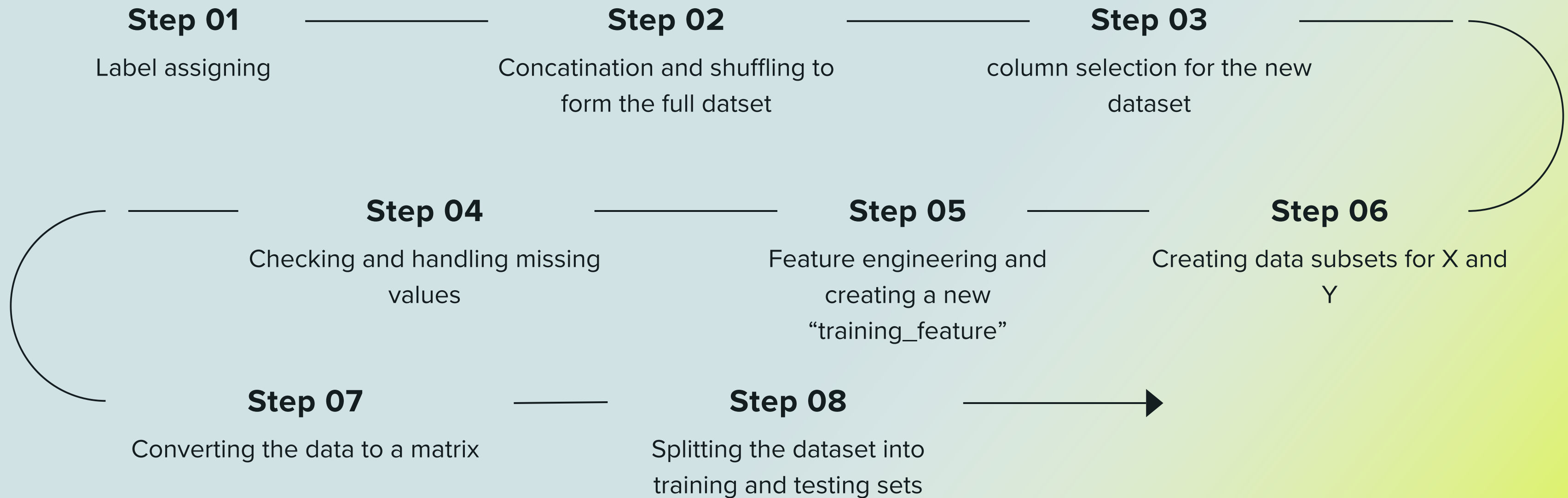


Fake News Detection Datasets

Kaggle is the world's largest data science community with powerful tool...

kaggle.com

Prétraitement



Fonction de Fitness

- La fonction de fitness est représentée par la valeur renvoyée par la fonction **objective_function**, dans ce cas c'est **“accuracy”**.
- La fonction **objective_function** prend un ensemble d'hyperparamètres (C et gamma) comme entrée, entraîne un modèle SVM avec ces hyperparamètres, et renvoie l'exactitude du modèle.
- L'ensemble du code est orienté vers **l'optimisation des hyperparamètres d'un SVM** en utilisant diverses métaheuristiques de jeux et d'algorithmes. Les résultats et performances des modèles sont évalués à chaque étape.

```
# Define the objective function to be optimized
def objective_function(x):
    #apply this one to execute mvpa and comment it to
    use the others
    svm = SVC(C=x["C"], gamma=x["gamma"],
kernel='rbf')
    #apply this one to execute two and hogo and
    comment it to use mvpa
    #svm = SVC(C=x[0], gamma=x[1], kernel='rbf')
    svm.fit(l_X_train, l_Y_train)
    y_pred = svm.predict(l_X_test)
    # Assuming binary classification, you can use F1
    score as the objective function
    accuracy = accuracy_score(l_Y_test, y_pred)

    return accuracy
```

FONCTION DE FITNESS

Implemenation avec les 4 metaheuristiques



8- Evaluations

Metric	ML - SVM	TOW	HOGO	LCA	MVPA
Accuracy	97%	97%	97%	/	97
F1	97%	97%	97%	/	97
Recall	97%	97%	97%	/	97
Precision	97%	97%	97%	/	97%

ça a juste monté en virgule a cause de la taille du subset pris