Optimisation Métaheuristique pour la Réorganisation de Textes Désorganisés : Compétition Santa 2024

Groupe

Imane Marrouss, Zineb Idabella , Anas Bouzina

Decembre 17, 2024

Introduction au Problème

La compétition Santa 2024 propose le défi suivant: réorganiser des mots désordonnés dans des textes pour minimiser leur perplexité.

La perplexité, une métrique couramment utilisée en traitement du langage naturel, mesure le degré de confusion qu'un lecteur ou un modèle ressent en lisant une séquence.

Une faible perplexité indique une séquence fluide et cohérente, essentielle pour la lisibilité et la compréhension.

Le problème est combinatoire par nature. Chaque texte désorganisé peut être représenté comme un ensemble de mots, et l'objectif est de trouver la permutation optimale qui minimise la perplexité.

Définition du problème d'optimisation : Kaggle Santa 2024

0.1 Fonction objective

La fonction objective de ce problème est de minimiser la perplexité des séquences proposées. En d'autres termes :

$$\text{Minimiser Perplexité}(S),$$

où S représente une séquence candidate obtenue en réorganisant les mots de la séquence initiale.

2. Fitness et Qualité des Solutions

La **fitness** d'une solution est définie comme l'inverse de la perplexité :

$$\text{Fitness}(S) = \frac{1}{Perplexité(S)}$$

- Une **solution de haute qualité** correspond à une séquence ayant une perplexité faible (fitness élevée), indiquant un texte bien structuré et cohérent.

- Une **solution de faible qualité** correspond à une séquence ayant une perplexité élevée (fitness faible), signalant un texte incohérent.

3. Nature combinatoire du problème

Le problème est combinatoire car il implique de trouver une permutation optimale des mots pour chaque séquence donnée.

Le nombre total de permutations possibles pour une séquence de n mots est donné par $n!$, ce qui rend l'espace de recherche extrêmement grand lorsque n augmente.

4. Complexité du problème

Le problème est **NP-difficile** pour les raisons suivantes :

1. **Espace de recherche exponentiel** : Avec $n!$ permutations possibles, le problème devient intractable pour des séquences longues.
2. **Réduction à un problème connu** : Le problème peut être réduit au **Problème d'Attribution Quadratique (QAP)**, un problème NP-difficile, où chaque permutation correspond à une attribution entre mots et positions.

5. Définition des solutions

Solution faisable : Une solution est une **permutation valide** des mots de la séquence donnée dans le fichier `sample_submission.csv`.

- **Conditions :**
  - Tous les mots doivent être présents.
  - Aucun mot ne peut être ajouté, supprimé ou modifié.
  - Chaque permutation doit être une réorganisation des mots de la séquence initiale.

Solution non faisable : Une solution est **non faisable** si elle ne respecte pas les contraintes ci-dessus, par exemple :

- Si des mots manquent ou sont répétés.
- Si des mots étrangers à la séquence sont introduits.

Espace de solutions : L'espace de solutions est l'ensemble de toutes les permutations possibles des mots d'une séquence donnée. Pour une séquence de n mots, l'espace de solutions est de taille $n!$.

6. Mouvements possibles (Voisinage)

Les mouvements possibles dans cet espace sont définis par les permutations des mots. Les plus courantes sont :

- **Échange de deux mots (Swap)** : $S = (w_1, w_2, ..., w_i, w_j, ..., w_n) \rightarrow S' = (w_1, w_2, ..., w_j, w_i, ..., w_n)$.
- **Inversion d'un sous-ensemble (Reverse)** : $S = (w_1, w_2, ..., w_i, ..., w_j, ..., w_n) \rightarrow S' = (w_1, w_2, ..., w_j, ..., w_i, ..., w_n)$.
- **Insertion** : Déplacer un mot à une autre position.

Ces mouvements permettent d'explorer l'espace de solutions et de converger vers une solution optimale.

Résumé : Définition formelle

- **Espace de solutions** : Ensemble des permutations de n mots ($n!$).
- **Solution faisable** : Permutation valide respectant les contraintes initiales.
- **Solution non faisable** : Toute permutation violant les contraintes.
- **Fonction objective** : Minimiser la perplexité des séquences.
- **Fitness** : Inverse de la perplexité.
- **Mouvements** : Échanges, inversions ou insertions dans les séquences.

Analyse et Compréhension du Problème

La compétition **Santa 2024** vise à réorganiser les mots désordonnés de textes pour minimiser la **perplexité**, une métrique évaluant la cohérence et la fluidité des séquences.

Données fournies :

- **Fichier :** `sample_submission.csv`
- **id :** identifiant unique.
- **text :** séquences de mots à réorganiser en permutations valides (aucun mot ajouté, supprimé ou modifié).

Évaluation :

- Le score est basé sur la **perplexité moyenne**, calculée par le modèle **Gemma 2 9B**, un modèle de langage avancé.
- Une faible perplexité indique des textes clairs et logiques.

Objectif :

Trouver une permutation optimale pour chaque séquence tout en respectant les contraintes, afin d'obtenir un score de perplexité minimal.

Métaheuristique Proposée : Approche Hybride ACO+SA

Pour résoudre le problème de minimisation de la perplexité, nous proposons une approche hybride combinant l'**Optimisation par Colonie de Fourmis (ACO)** et le **Recuit Simulé (SA)**.

Cette méthode exploite les forces complémentaires des deux techniques pour explorer efficacement l'espace des permutations et affiner les solutions.

Comment fonctionne l'approche hybride ?

1. Entrées de l'approche hybride

- **Solution initiale :** Chaque séquence initiale de n mots. Les solutions initiales sont des permutations aléatoires ou basées sur une heuristique simple (par exemple, l'ordre alphabétique ou selon la fréquence).
- **Espace des solutions :** Toutes les permutations possibles de n mots ($n!$ permutations).
- **Opérateurs de mouvements :**
  - Pour ACO : Transition probabiliste entre les nœuds (mots) influencée par les phéromones.
  - Pour SA : Modifications locales (échanges de mots, inversions de sous-séquences, etc.).
- **Paramètres principaux :**
  - Pour ACO :
    - α : Influence des phéromones sur les probabilités de transition.
    - β : Influence des heuristiques locales sur les probabilités de transition.
    - ρ : Taux d'évaporation des phéromones.
    - m : Nombre de fourmis.
  - Pour SA :
    - T_0 : Température initiale.
    - λ : Taux de décroissance de la température.
    - k : Nombre d'itérations par température.
- **Contrôles d'intensification et de diversification :**
  - **Intensification :** Concentration des phéromones sur les solutions prometteuses dans ACO.
  - **Diversification :** Exploration de voisinages variés dans SA et évaporation des phéromones pour éviter les minima locaux.

2. Phase 1 : Construction des solutions avec ACO

Objectif : Générer des permutations valides influencées par des phéromones et des heuristiques locales.

1. **Représentation sous forme de graphe :**
   - Chaque mot est un nœud.
   - Les arêtes représentent les transitions possibles entre les mots, pondérées par des phéromones (τ_{ij}) et une heuristique locale (η_{ij}), comme l'association lexicale ou les probabilités contextuelles.
2. **Mécanisme de construction :**
   - Chaque fourmi commence sur un nœud aléatoire.
   - À chaque étape, elle choisit le prochain mot à ajouter à la permutation en fonction de la probabilité suivante :

$$P_{ij} = \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{k \in \text{non visités}} \tau_{ik}^{\alpha} \cdot \eta_{ik}^{\beta}}.$$

3. **Critère de sélection :**
   - Après chaque itération, les meilleures permutations sont sélectionnées selon leur perplexité.
4. **Mise à jour des phéromones :**
   - Les phéromones sur les arêtes correspondant aux solutions de haute qualité sont augmentées ($\Delta\tau \propto \frac{1}{\text{Perplexité}(S)}$).
   - Une évaporation partielle réduit les phéromones ($\tau_{ij} = (1-\rho)\tau_{ij}$) pour favoriser la diversification.

3. Phase 2 : Affinage des solutions avec SA

Objectif : Affiner les permutations générées par ACO pour réduire davantage la perplexité.

1. **Solution initiale :** Les meilleures permutations issues de la phase ACO.

2. **Exploration des voisinages :**
   - Les opérateurs de mouvements incluent :
     - Échange de deux mots (swap).
     - Inversion d'une sous-séquence.
     - Déplacement d'un mot à une autre position.
   - Un voisin est généré aléatoirement à chaque étape.
3. **Critère d'acceptation :**
   - Une solution avec une meilleure perplexité est toujours acceptée.
   - Une solution moins bonne est acceptée avec une probabilité donnée par :

     $$P = e^{-\Delta\text{Perplexité}/T},$$

     où ΔPerplexité est la différence de perplexité entre la nouvelle solution et l'ancienne, et T est la température actuelle.
4. **Refroidissement :**
   - La température est mise à jour après un nombre fixé d'itérations ($T \leftarrow \lambda T$).
5. **Critère d'arrêt :**
   - Le processus s'arrête lorsque la température atteint un seuil minimal ou après un nombre maximal d'itérations.

4. Interaction entre ACO et SA

Objectif : Utiliser les avantages complémentaires des deux approches pour une optimisation efficace.

1. Les solutions améliorées par SA sont utilisées pour mettre à jour les phéromones dans ACO, ce qui favorise les trajectoires de haute qualité.
2. Les phases ACO et SA sont répétées plusieurs fois (par exemple, jusqu'à convergence ou une limite de temps).
3. La dernière solution est celle qui a la perplexité minimale parmi toutes les solutions explorées.

5. Résumé des étapes de l'approche hybride

1. Générer des permutations initiales à l'aide d'ACO.
2. Affiner chaque permutation via SA pour améliorer sa qualité.
3. Mettre à jour les phéromones en fonction des solutions issues de SA.
4. Répéter les phases jusqu'à atteindre un critère d'arrêt.

6. Avantages de cette approche

- Équilibre entre exploration globale (ACO) et exploitation locale (SA).
- Réduction efficace de la perplexité grâce à une combinaison stratégique des deux techniques.
- Adaptabilité et robustesse pour des problèmes combinatoires complexes.