Cahier des Charges - Projet AMS

1. Contexte et Objectifs du Projet

1.1. Contexte

Le projet s'inscrit dans le cadre de l'analyse des relations narratives au sein de textes littéraires, spécifiquement la saga Fondation d'Isaac Asimov. Il vise à extraire des entités nommées (personnages) et à construire un réseau relationnel entre elles.

Ce projet répond à un besoin croissant de visualiser les interactions narratives dans les grandes œuvres littéraires pour des applications telles que l'analyse littéraire, la génération de résumés et l'exploration des relations narratives.

1.2. Objectifs

- Développer un pipeline complet pour l'extraction des personnages et la génération de graphes relationnels.
- Améliorer la précision de l'extraction des entités nommées et des relations en utilisant des modèles hybrides (symboliques, probabilistes, et IA).
- Tester et valider les graphes générés avec des utilisateurs experts ou des outils d'analyse littéraire.

---

2. Périmètre du Projet

2.1. Fonctionnalités attendues

- Extraction d'entités nommées (NER) de type "personnage".
- Filtrage et validation des entités pour éviter les faux positifs.
- Gestion des alias et regroupement des variantes des noms de personnages.
- Création d'un graphe relationnel représentant les interactions entre personnages.
- Export des graphes au format `GraphML`.

2.2. Technologies utilisées

- **Langages :** Python 3.
- **Bibliothèques :**
    - `stanza` : traitement linguistique (tokenisation, POS tagging, NER).
    - `flair` : modèle "ner-french" pour l'extraction d'entités.
    - `networkx` : gestion et export des graphes.
    - `pandas` : manipulation et structuration des données (export CSV).

- **Plateformes :** Compatible avec Linux/Windows pour exécuter les scripts Python.

3. Public Cible

3. Profil des utilisateurs

- Professionnels ou chercheurs en littérature.
- Développeurs et analystes travaillant sur des projets d'analyse de texte ou NLP.
- Compétences techniques : connaissances de base en Python et manipulation de données textuelles.

4. Scénarios de Tests Utilisateurs

4.1. Objectifs des tests utilisateurs

- Valider la précision de l'extraction des personnages.
- Vérifier la pertinence des relations établies entre personnages.
- Tester l'export des graphes et leur visualisation.

4.2. Méthodologie de tests

- Tests supervisés (présence d'un facilitateur pour guider l'utilisateur).
- Utilisation d'exemples issus des textes d'Isaac Asimov.
- Évaluation des résultats via des questionnaires qualitatifs et des métriques quantitatives (e.g., taux de faux positifs, complétude des relations).

4.3. Scénarios de tests

- Extraction des personnages d'un chapitre donné.
- Identification et suppression des entités parasites (e.g., lieux mal classifiés comme personnages).
- Génération d'un graphe et validation de sa structure.
- Export du graphe au format `GraphML`.

5. Critères de Réussite et Évaluations

5.1. Indicateurs de performance (KPI)

- Taux de détection correcte des personnages (>90%).
- Taux d'erreurs dans l'association des relations (<40%).
- Temps moyen pour traiter un chapitre (<5 minutes).
- Satisfaction des utilisateurs (notation >4/5).

5.2. Feedback des utilisateurs

- Collecte de retours via des questionnaires standardisés après chaque test.
- Analyse qualitative des commentaires pour identifier les améliorations possibles.

6. Contraintes et Risques

6.1. Contraintes techniques

- Performances des modèles NLP pour le français (limité par les modèles disponibles).
- Gestion de grandes quantités de texte (mémoires et temps de traitement).
- Compatibilité multi-plateformes.

6.2. Risques potentiels

- Faux positifs dans l'extraction des entités.
- Alias mal gérés conduisant à des regroupements incorrects.
- Délais liés à l'optimisation des performances du pipeline.

7. Méthodologie

Le projet suit les étapes suivantes :

1. **Nettoyage des textes :** Suppression des caractères inutiles et normalisation des entités (e.g., conversion en minuscules, suppression des retours à la ligne).
2. **Extraction des entités nommées :** Utilisation de modèles NLP (`stanza` et `flair`) pour détecter les personnages et lieux tout en filtrant les entités non pertinentes.
3. **Gestion des alias :** Regroupement des variantes d'un même nom (e.g., noms de famille communs, prénoms isolés) pour éviter les doublons dans le graphe.
4. **Calcul des relations :** Analyse des co-occurrences entre personnages dans un rayon défini (25 tokens) pour établir des liens relationnels.

5. **Construction du graphe :** Génération d'un graphe relationnel avec `networkx`, ajout des nœuds (personnages) et arêtes (relations), et calcul des poids des relations.
6. **Export des résultats :** Conversion des graphes au format `GraphML` pour une visualisation et une analyse ultérieures.

---

8. Planning Prévisionnel

8.1. Phases du projet

1. **Phase de conception :** 2 semaines.
2. **Phase de développement :** 4 semaines.
3. **Phase de tests utilisateurs :** 2 semaines.
4. **Ajustements post-tests :** 2 semaines.

8.2. Livrables

- Prototype fonctionnel pour l'extraction des personnages.
- Script de génération et d'export des graphes.
- Rapport des tests utilisateurs.
- Documentation utilisateur et technique.

9. Équipe du Projet

- **Chef de projet :** Juan-Manuel Torres.
- **Développeurs principaux :** Ylies Chementel, Mohamed Rouabhia.

10. Annexes

- Exemples de textes traités.
- Exemple de graphe exporté au format `GraphML`.
- Liste des outils et modèles utilisés (avec documentation associée).