T-ESP-700

Big Brain

Lucas FIXARI
Pierre ROCHETTE
William WOZIWODA

Présentation de l'IA d'Analyse Documentaire

1. Description du Projet

Titre : IA d'Analyse Documentaire Locale pour la Recherche Contextuelle en Langage Naturel

Description : Une IA locale capable d'analyser, indexer et rechercher des informations à partir de divers documents. Conçue pour assurer la confidentialité, elle permet une recherche intuitive en langage naturel et fournit des réponses contextualisées avec les sources documentaires.

2. Contexte du Projet

Dans un monde où le numérique prend une place croissante et où le coût de stockage diminue, **les particuliers comme les organisations accumulent des quantités massives de données documentaires : rapports, contrats, relevés, images scannées, et bien plus encore**. Cette transition vers le format numérique, motivée par des raisons économiques et écologiques, s'accélère, reléguant le papier au second plan. Pourtant, **accéder à une information spécifique au sein de cet ensemble peut vite devenir un défi**, entraînant des pertes de temps et de ressources.
Les entreprises, les administrations, et même les particuliers sont régulièrement confrontés à ce besoin d'**accès rapide et fiable aux informations sans avoir à connaître la source exacte du document ou son emplacement**. *Qui n'a jamais rêvé de retrouver facilement une fiche d'imposition d'il y a plusieurs années sans devoir fouiller dans trois classeurs ? Et si ce besoin ne vous parle pas, demandez donc à vos parents.*
Ce projet vise à combler cette lacune avec une solution innovante : une **intelligence artificielle locale et autonome, capable de lire, analyser, et indexer des documents variés** (PDF, DOCX, Excel, images, etc.) et de fournir des réponses instantanées à des **questions formulées en langage naturel**.
Contrairement aux solutions basées sur le cloud, qui posent des questions de sécurité et de confidentialité des données, cette IA locale **répond à des enjeux de confidentialité tout en s'adaptant aux contraintes des infrastructures internes des utilisateurs**.
L'objectif final est de proposer un outil performant et facile d'utilisation, permettant aux utilisateurs d'**obtenir des réponses précises à des questions d'ordre thématique ou chiffré**, en ayant simplement **connaissance de certains mots-clés ou d'un thème général**. Ce projet s'inscrit dans la mouvance actuelle de l'automatisation des processus documentaires et de la démocratisation de l'IA pour des applications pratiques au quotidien. Il répond ainsi à des besoins concrets d'optimisation, de confidentialité et de productivité.

3. Objectifs du Projet

Objectifs principaux

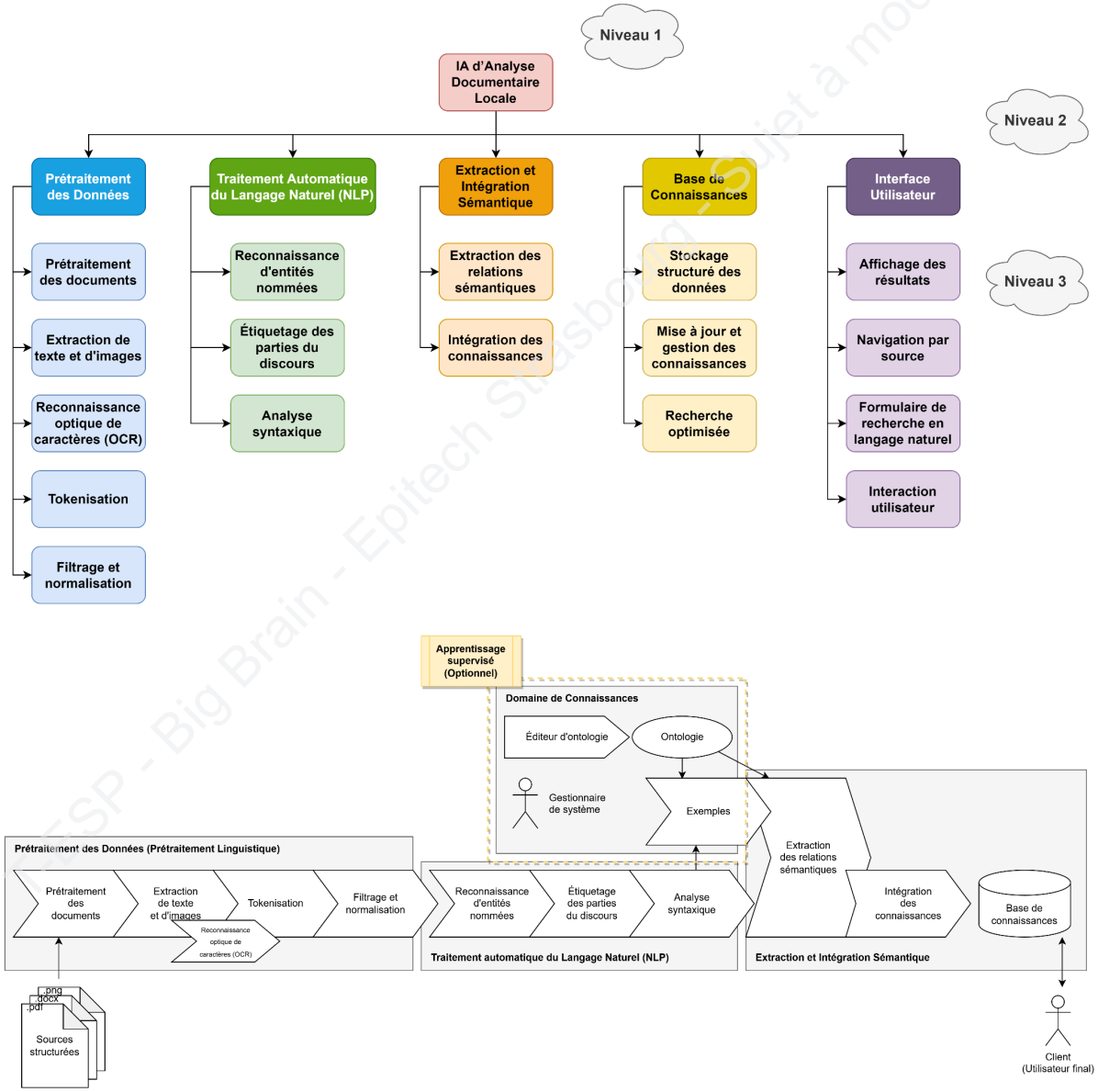
- **Confidentialité et Sécurité** : Assurer un traitement local de l'information pour une sécurité accrue, adapté aux entreprises, administrations, et particuliers.
- **Accessibilité des Informations** : Permettre une recherche simplifiée et instantanée sans connaissance préalable des fichiers sources.

- **Efficacité et Précision** : Délivrer des réponses claires et pertinentes en fonction des questions posées, avec des résumés et des sources explicites.

Objectifs techniques

- Développer une IA (en Python) pour l'analyse et l'indexation des documents (PDF, DOCX, Excel, images, etc).
- Créer une interface simple pour poser des questions en langage naturel et visualiser les résultats.
- Optimiser la performance pour une utilisation fluide en local.

4. Structure de Décomposition des Produits (PBS)



Niveau 1 : Solution IA d'Analyse Documentaire Locale

- **Objectif** : Développement complet d'une solution d'analyse documentaire pour une recherche contextuelle rapide et sécurisée, adaptée à un environnement local.

Niveau 2 : Modules Principaux

1. **Prétraitement des Données**
    - **Objectif** : Préparer les documents pour les étapes de traitement NLP et d'indexation en assurant un format de données homogène.
2. **Traitement Automatique du Langage Naturel (NLP)**
    - **Objectif** : Comprendre et analyser le contenu des documents de manière contextuelle, en extrayant les entités et relations sémantiques.
3. **Domaine de Connaissances (Optionnel)**
    - **Objectif** : Enrichir le traitement avec une ontologie et des exemples pour une analyse plus approfondie, si nécessaire.
4. **Extraction et Intégration Sémantique**
    - **Objectif** : Extraire les relations sémantiques et intégrer les informations extraites dans une base de connaissances structurée.
5. **Base de Connaissances**
    - **Objectif** : Stocker et organiser les informations pour une recherche rapide et fiable.
6. **Interface Utilisateur**
    - **Objectif** : Fournir une interface interactive pour permettre à l'utilisateur de rechercher et d'afficher les résultats de manière intuitive.

Niveau 3 : Tâches Spécifiques

1. **Prétraitement des Données**
    - **Prétraitement des documents** : Nettoyage de texte, suppression de caractères spéciaux, conversion en texte brut, détection de la langue.
    - **Extraction de texte et d'images** : Utilisation de bibliothèques pour extraire le texte et les images des documents (PDF, DOCX, et XLSX).
    - **Reconnaissance optique de caractères (OCR)** : Utilisation de l'OCR (Tesseract) pour extraire le texte des images (PNG, JPG) scannées.
    - **Tokenisation** : Division du texte en unités analytiques (tokens) pour faciliter l'analyse NLP.
    - **Filtrage et normalisation** : Suppression des stop words et lemmatisation pour normaliser le texte.
2. **Traitement Automatique du Langage Naturel (NLP)**
    - **Reconnaissance d'entités nommées** : Identification des entités clés comme les noms, dates, lieux, etc.
    - **Étiquetage des parties du discours (POS Tagging)** : Analyse grammaticale pour identifier la fonction de chaque mot dans la phrase.
    - **Analyse syntaxique** : Détermination de la structure des phrases pour comprendre les relations entre mots.
    - ***Interprétation des questions (au besoin)*** : Identification des intentions dans les requêtes de l'utilisateur pour orienter la recherche.

3. **Domaine de Connaissances (Optionnel)**
    - **Éditeur d'ontologie** : Création et gestion d'une ontologie pour structurer les connaissances et relations entre concepts.
    - **Ontologie** : Définition de catégories et relations thématiques pour organiser les informations de manière hiérarchique.
    - **Exemples** : Utilisation d'exemples spécifiques pour guider l'apprentissage supervisé, si applicable.
4. **Extraction et Intégration Sémantique**
    - **Extraction des relations sémantiques** : Identification des relations entre les entités pour offrir des réponses contextuelles précises.
    - **Intégration des connaissances** : Structuration et stockage des relations extraites dans un format indexé pour faciliter la recherche.
5. **Base de Connaissances**
    - **Stockage structuré des données** : Utilisation de bases de données comme SQLite pour stocker les informations extraites et indexées.
    - **Mise à jour et gestion des connaissances** : Mécanisme de mise à jour des données et gestion des erreurs pour garantir une base de connaissances fiable et à jour.
    - **Recherche optimisée** : Indexation des mots-clés et métadonnées pour des recherches efficaces et rapides.
6. **Interface Utilisateur**
    - **Affichage des résultats** : Interface de visualisation des réponses, incluant les résumés et les extraits pertinents.
    - **Navigation par source** : Possibilité pour l'utilisateur de consulter les documents sources et d'explorer par thème ou type de document.
    - **Formulaire de recherche en langage naturel** : Interface permettant de poser des questions en langage naturel.
    - **Interaction utilisateur** : Interaction en temps réel pour des recherches rapides et une restitution de résultats intuitive.

5. Description Technologies du Logiciel

Technologies par Module

1. **Prétraitement des Données**

Prétraitement des documents

- **Technologies** : *re* (expressions régulières pour le nettoyage du texte), *unicodedata* (normalisation du texte).

Extraction de texte et d'images

- **Technologies** :
    - **PDF** : *PyMuPDF* pour extraire le texte des fichiers PDF.
    - **DOCX** : *python-docx* pour extraire le texte des fichiers Word.

- **XLSX** : *openpyxl* pour extraire le contenu des fichiers Excel.

Reconnaissance optique de caractères (OCR)

- **Technologie** : *pytesseract* pour extraire le texte des images scannées.

Tokenisation

- **Technologie** : *nltk.tokenize* ou *spaCy* pour diviser le texte en tokens.

Filtrage et normalisation

- **Technologies** : *nltk.corpus.stopwords* pour supprimer les mots communs, *spaCy* pour la lemmatisation et la normalisation.

2. **Traitement Automatique du Langage Naturel (NLP)**

Reconnaissance d'entités nommées

- **Technologies** : *spaCy* pour la reconnaissance d'entités nommées, ou *Transformers* (via Hugging Face) pour des modèles avancés comme BERT.

Étiquetage des parties du discours (POS Tagging)

- **Technologies** : *spaCy* pour l'étiquetage des parties du discours.

Analyse syntaxique

- **Technologies** : *spaCy* pour la dépendance syntaxique, *Transformers* pour des modèles NLP plus avancés.

3. **Domaine de Connaissances (Optionnel)**

Éditeur d'ontologie

- **Technologies** : *rdflib* pour créer et gérer des ontologies en RDF, ou un éditeur personnalisé en Python.

Ontologie

- **Technologie** : *rdflib* pour définir et gérer une ontologie sous forme de graphe de connaissances.

Exemples

- **Technologie** : *scikit-learn* ou *Transformers* pour créer des ensembles d'exemples en apprentissage supervisé, si nécessaire.

4. **Extraction et Intégration Sémantique**

Extraction des relations sémantiques

- **Technologies** : *spaCy* pour la détection des relations entre entités, ou *Transformers* pour les modèles capables de détecter des relations sémantiques.

Intégration des connaissances

- **Technologies** : *networkx* pour structurer les connaissances sous forme de graphe, *SQLite* pour stocker les relations sous forme d'index.

5. **Base de Connaissances**

Indexation des contenus et des métadonnées

- **Technologies** : *SQLite* pour la base de données locale, *Whoosh* pour créer un moteur de recherche en texte intégral.

Structuration des données

- **Technologies** : *SQLite* pour structurer et gérer les données de manière hiérarchique.

6. **Interface Utilisateur**

Compréhension des requêtes

- **Technologies** : *spaCy* pour l'analyse des intentions, *Transformers* pour la compréhension approfondie des questions.

Génération de résumés

- **Technologies** : *Sumy* pour les résumés simples, ou *Transformers* avec des modèles de type T5 ou BART pour des résumés avancés.

Interface d'interaction

- **Technologies** : *Flask*, *FastAPI*, ou *tkinter* pour construire l'interface utilisateur permettant de poser des questions et d'afficher les résultats.

6. Plan de Gestion du Projet

Phases de Gestion et Suivi

1. **Planification Initiale** : Estimation du temps et des ressources pour chaque module.
2. **Contrôle Qualité** : Vérification des livrables pour assurer la précision et la pertinence des résultats.
3. **Gestion des Risques** : Identification des risques pour chaque phase et plan de contournement.
4. **Suivi et Reporting** : Rapports d'avancement mensuels incluant :
    - Ce qui a été réalisé
    - Difficultés rencontrées
    - Suivi des KPIs (ex. rapidité de l'analyse, précision des réponses)

7. Prototypage et Tests

Prototypes à Développer

1. **Module d'Analyse des Documents** : Tester l'extraction et la conversion de texte.
2. **Module d'Indexation** : Vérifier la rapidité et précision des recherches.
3. **Module de Langage Naturel** : Assurer la compréhension des requêtes utilisateur.
4. **Synthèse et Interface Utilisateur** : Prototyper une interface de questions/réponses.

Plan de Test

- **Qualité des résultats** : Mesurer la précision des réponses pour les requêtes basiques.
- **Performance** : Temps de réponse pour l'indexation et la recherche.
- **Expérience Utilisateur** : Test d'interface pour une navigation fluide.