Système de Gestion Intelligente des Documents

Documentation Technique et Guide Utilisateur

Table des Matières

1. Introduction

Le Système de Gestion Intelligente des Documents est une application Streamlit conçue pour automatiser le traitement, l'analyse et la classification des documents numérisés. Il combine OCR (Reconnaissance Optique de Caractères), traitement d'image et techniques de NLP pour offrir une solution complète de gestion documentaire.

1.1 Caractéristiques Principales

- Traitement d'images et OCR multilingue
- Génération automatique de tags
- Classification automatique des documents
- Interface utilisateur intuitive
- Analyse statistique et visualisation
- Export des données dans plusieurs formats

2. Installation

2.1 Prérequis

- Python 3.8 ou supérieur

- Tesseract OCR
- Poppler (pour le traitement PDF)

2.2 Installation des Dépendances

```
pip install streamlit
pip install pytesseract
pip install pdf2image
pip install opencv-python
pip install pandas
pip install numpy
pip install scikit-learn
pip install pillow
pip install seaborn
pip install matplotlib
pip install nltk
pip install rake-nltk
```

2.3 Configuration de Tesseract

1. Installer Tesseract OCR sur votre système
2. Ajouter le chemin de Tesseract aux variables d'environnement
3. Vérifier l'installation : `tesseract --version`

3. Architecture du Système

3.1 Structure des Fichiers

```
project/
├── app.py              # Application principale
├── requirements.txt    # Dépendances
└── assets/            # Ressources statiques
```

3.2 Composants Principaux

1. **Interface Utilisateur (Streamlit)**
    - Gestion des formulaires
    - Affichage des visualisations
    - Interaction utilisateur
2. **Traitement d'Image**

    - Prétraitement des images
    - Amélioration de la qualité
    - OCR et extraction de texte

3. **Traitement du Langage**

    - Génération de tags
    - Classification des documents
    - Analyse textuelle

4. **Gestion des Données**

    - Stockage en session
    - Export des données
    - Statistiques et analyses

4. Fonctionnalités

4.1 Traitement d'Image

- Ajustement de la luminosité (-100 à +100)
- Contrôle du contraste (0.0 à 3.0)
- Réduction du bruit (flou gaussien)
- Prévisualisation en temps réel

4.2 OCR et Extraction de Texte

- Support multilingue (français et anglais)
- Prétraitement automatique
- Optimisation de la reconnaissance

4.3 Génération de Tags

- Extraction par RAKE
- Analyse TF-IDF
- Détection des noms propres
- Filtrage intelligent des stopwords

4.4 Classification

- Apprentissage automatique (Naive Bayes)
- Classification multiclasse
- Scores de confiance

4.5 Analyse et Statistiques

- Distribution des catégories
- Timeline des documents
- Statistiques textuelles
- Visualisations interactives

5. Guide Utilisateur

5.1 Import de Documents

1. Accéder à l'onglet "Import et OCR"
2. Télécharger un document (PDF ou image)
3. Ajuster les paramètres de traitement si nécessaire
4. Cliquer sur "Appliquer le traitement"
5. Extraire le texte

5.2 Gestion des Tags

1. Utiliser la génération automatique de tags
2. Sélectionner les tags pertinents
3. Ajouter des tags manuels si nécessaire
4. Valider la sélection

5.3 Classification et Analyse

1. Entraîner le classificateur
2. Visualiser les statistiques
3. Effectuer des recherches
4. Exporter les données

6. API et Classes

6.1 Classe Document

```
class Document:
    def __init__(self, name, content, category=None, metadata=None):
        self.name = name
        self.content = content
        self.category = category
        self.metadata = metadata or {}
        self.timestamp = datetime.now()
        self.confidence_score = None
```

6.2 Fonctions Principales

```
def preprocess_image(image, brightness=0, contrast=1.0, blur_amount=0)
def generate_tags(text, num_tags=10)
def extract_text_from_image(image)
def save_document(file, text_content, category, tags, processing_params=None)
```

7. Maintenance et Dépannage

7.1 Problèmes Courants

1. **Erreur OCR**
    - Vérifier l'installation de Tesseract
    - Ajuster les paramètres de prétraitement
    - Vérifier la qualité de l'image source
2. **Génération de Tags**
    - Vérifier la qualité du texte extrait
    - Ajuster le nombre de tags demandés
    - Vérifier les ressources NLTK
3. **Performance**
    - Optimiser la taille des images
    - Limiter le nombre de documents en session
    - Nettoyer régulièrement le cache

7.2 Maintenance

- Mettre à jour les dépendances régulièrement
- Sauvegarder les données importantes
- Monitorer l'utilisation des ressources

7.3 Bonnes Pratiques

- Prétraiter les images avant OCR
- Valider les tags générés
- Effectuer des sauvegardes régulières
- Maintenir une nomenclature cohérente

Contact et Support

Pour toute question ou support technique, veuillez contacter l'équipe de développement.

Note: Cette documentation est un document vivant qui sera mis à jour régulièrement pour refléter les évolutions du système.