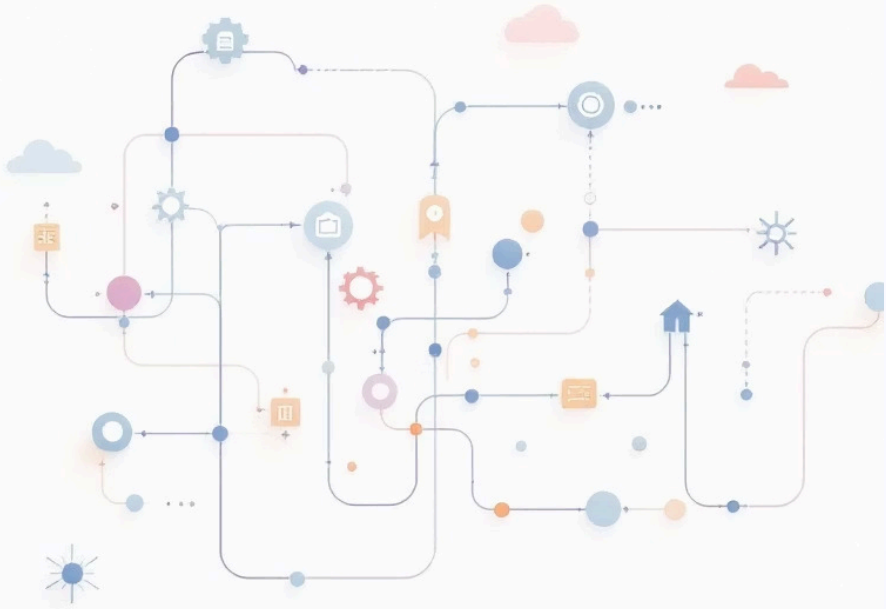


Automatisation d'Export et Classification de PDFs

Pipeline complet pour l'extraction, la vérification et le traitement OCR des documents HAL-INRIA (RRT) - disponible sur github datalake: **https://github.com/INRIA/datalake**

Présenté par Andréa NEBOT

Vue d'ensemble

Trois Étapes d'Automatisation

01

Export et Vérification

Téléchargement intelligent depuis l'API HAL avec validation des métadonnées

02

Classification OCR

Tri automatique des documents océrisés et non-océrisés

03

Conversion Texte

Traitement OCR des fichiers image vers format texte interrogeable

Étape 1 : Export des Fichiers

Interrogation API HAL

Requêtes ciblées vers la collection INRIA-RRRT

- ID HAL, titre, date
- Nombre de pages annoncées
- Liens PDF associés

Téléchargement Intelligent

Gestion Automatique des Fichiers

Téléchargement Structuré

Organisation par année dans `./output/liste_pdf_rrt/{année}/`

Évitement Doublons

Vérification pré-téléchargement pour ignorer les fichiers existants

Résumé CSV

`telechargements_par_annee.csv` avec comptage automatique

Vérification Pages PDF

1

Extraction Métadonnées

Nombre de pages depuis `page_s` API HAL

2

Comptage Réel

Analyse du PDF téléchargé

3

Comparaison

Colonne `pages_match` True/False

Sortie : `verifications_pages.csv` avec statut de correspondance

Gestion des Divergences

Isolation et Reporting

Fichiers Non-Conformes

Copie automatique vers :

`output/divergences_pages/{année}/`

Rapports Générés

- **HTML interactif** : tableau stylisé avec codes couleurs
- **PDF imprimable** : version rapport identique
- Statistiques : total, divergences, taux correspondance

Étape 2 : Classification OCR

Analyse du Dossier

Parcours de `liste_pdf_1980_1990` pour détecter tous les PDFs

Création Architecture

RRT_OCR : fichiers texte océrisés

RRT_SANS_OCR : fichiers images uniquement

Tri Automatique

Classification selon présence ou absence de texte extractible

Détection OCR : Comment ça Marche ?

Analyse Textuelle

Extraction tentative du contenu texte du PDF

Classification

Présence texte → OCR

Absence texte → Image

✓ Fichiers avec OCR

Texte interrogeable, recherche possible, métadonnées préservées

□ Fichiers sans OCR

Images scannées, texte non-extractible, nécessite conversion

Distribution des Fichiers OCR et Non-OCR par Année

Ce graphique illustre l'évolution du nombre de documents classifiés comme "océrisés" ou "sans OCR" au fil des ans. On observe un changement significatif aux alentours du milieu des années 90.

1

Avant 1994

la majorité des documents étaient des images numérisées, donc "sans OCR".

2

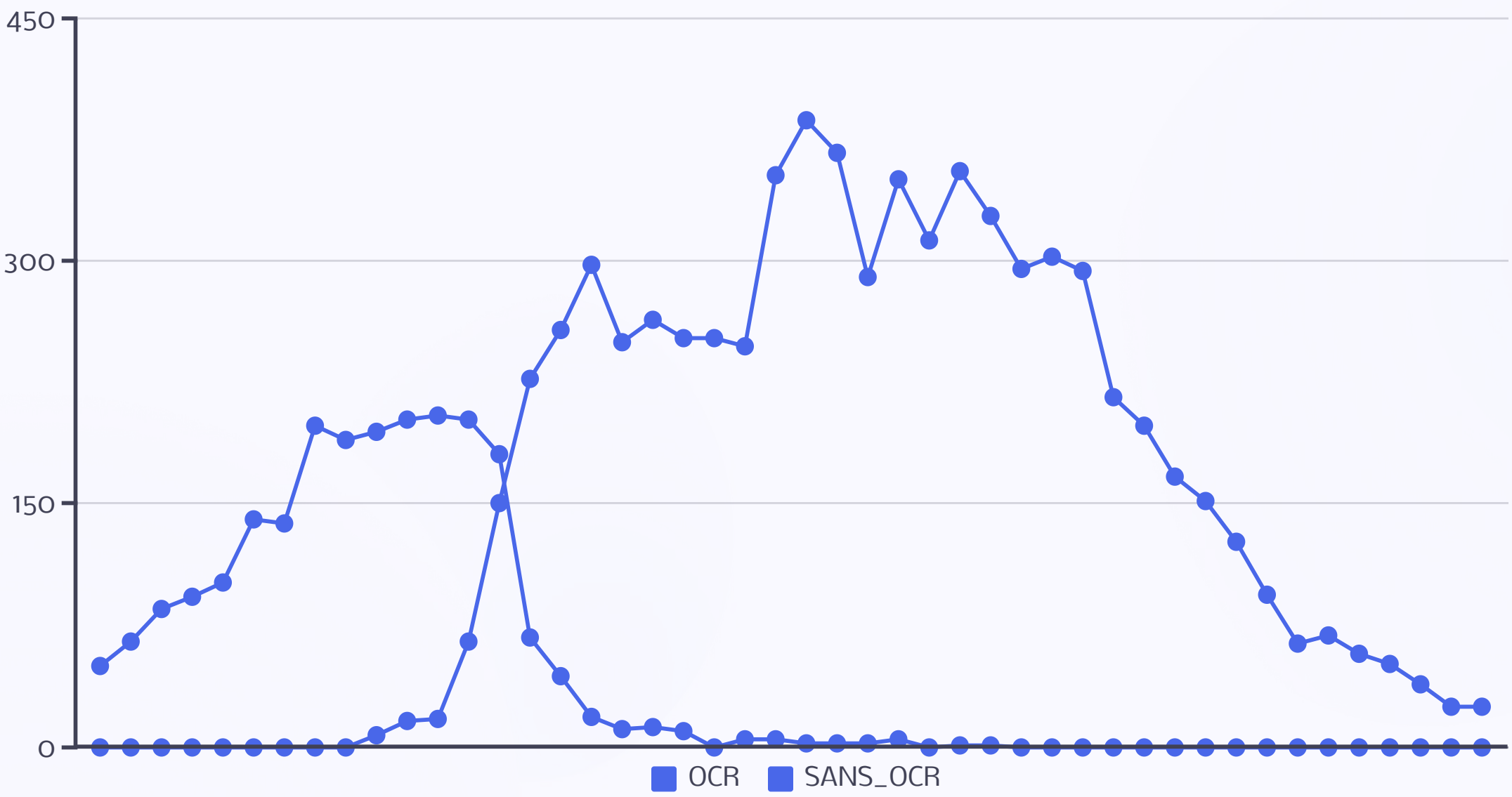
Après 1994

avec l'amélioration des technologies, la proportion de documents "océrisés" a considérablement augmenté, indiquant une meilleure préparation pour l'extraction de texte.

3

La colonne "ERREUR"

est constamment à zéro, ce qui signifie que notre pipeline de classification est stable et ne génère pas d'erreurs de statut.



Étape 3 : Conversion OCR

1. **Lecture PDF**
   Dossier `RRT_SANS_OCR`
2. **Conversion Image**
   Chaque page → format image
3. **OCR Tesseract**
   Extraction texte intelligente
4. **Sauvegarde TXT**
   Dossier RRT_OCR_FROM_IMG

Installation Requise

Dépendances Python

```
pip install ocrmypdf pytesseract pdf2image pillow
```

ocrmypdf

Wrapper Python pour OCR automatique de PDFs

pytesseract

Interface Python pour moteur OCR Tesseract

pdf2image

Conversion pages PDF en images PIL

pillow

Bibliothèque traitement d'images Python

Prêt à automatiser votre pipeline documentaire !