Numériser des fiches de retour d'expérience sur le développement des lanceurs spaciaux

ScanRexs, un OCR LSTM pour la numérisation des FPCs
Vers la détection des signaux faibles dans le corpus

Elvis MBONING, Nadège LECHEVREL
Michal KURELA, Damien NOUVEL

CNES — INALCO ERTIM

SIFED-Tours — 31 mai 2018

Plan

Contexte

- Détection des **signaux faibles** pour la **prévention des risques**
- Solution **TAL** pour automatiser cette tâche de détection
- Méthodes statistiques pour la **détection** et la **classification** des signaux faibles dans les REX [Galand, 2017]
- Priorité de la numérisation optimale des FPC
- Deep Learning (OCR/LSTM) pour la numérisation des documents hétérogènes

Résumé de l'étude

- Extraction de contenus textuels avec **ScanRexs**
  - Analyse et prétraitement des données au processus d'OCR
  - Prise en compte des propriétés internes des FPC (filtres)
  - Procédure de conversion de PDF au TIFF spécifique aux FPC
  - Binarisation, segmentation
  - Reconnaissance, visualisation, édition
  - Annotation, étude terminologique
- Génération de modèles optimisés de reconnaissance LSTM
  - Annotation et extraction
  - Entraînement, évaluation, optimisation
- Numérisation automatisée de milliers de FPC mises à notre disposition

Plan

Extraction d'information : Méthodologie récursive

Données en entrée : fichiers PDF (FPC) hétérogènes

Sortie voulue : fichiers texte brut

- Organisation des fichiers disponibles (images vs texte)
- Traitement des PDF images, conversion en TIFF
- Adaptation d'un outil OCR libre (Ocropy)
- Application de l'OCR à partir d'un modèle par défaut (anglais)
- Annotation, entraînement, évaluation, optimisation
- …

Plan

Corpus et caractéristiques des fichiers FPC I

- Corpus disponible
  - **516 PDF**
  - **Multilingue** : fr / en / es
  - **Hétérogénéité** : polices, qualité, dates des fiches

Compression PDF images		
Filter : /CCITTFaxDecode	117 – 120	310
Filter : /JBIG2Decode	0 – 300	83
Filter : /DCTDecode	0 – 300	5
Filter : /FlateDecode	0 – 300	110
Filter : [’/FlateDecode’, ’/DCTDecode’]	0 – 300	8

TABLE – Filtres utilisés dans les PDF pour encapsuler les images numérisées (les valeurs à droite représentent le nombre de filtres d’objets requis pour le décodage de l’image encapsulée)

Solution de conversion des fichiers FPC

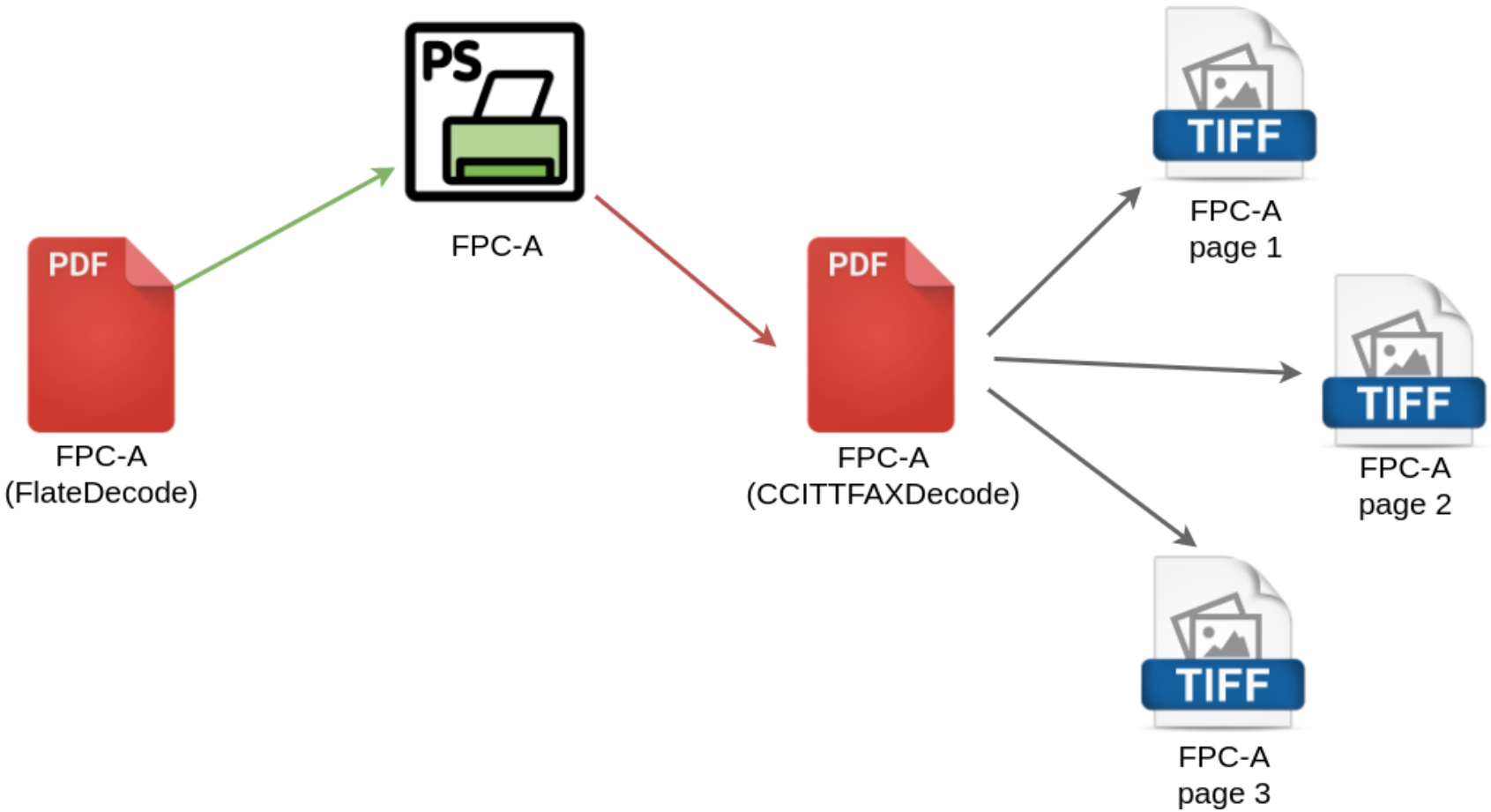


FIGURE – Solution pour résoudre le décodage de FlateDecode en CCITTFAXDecode

Plan

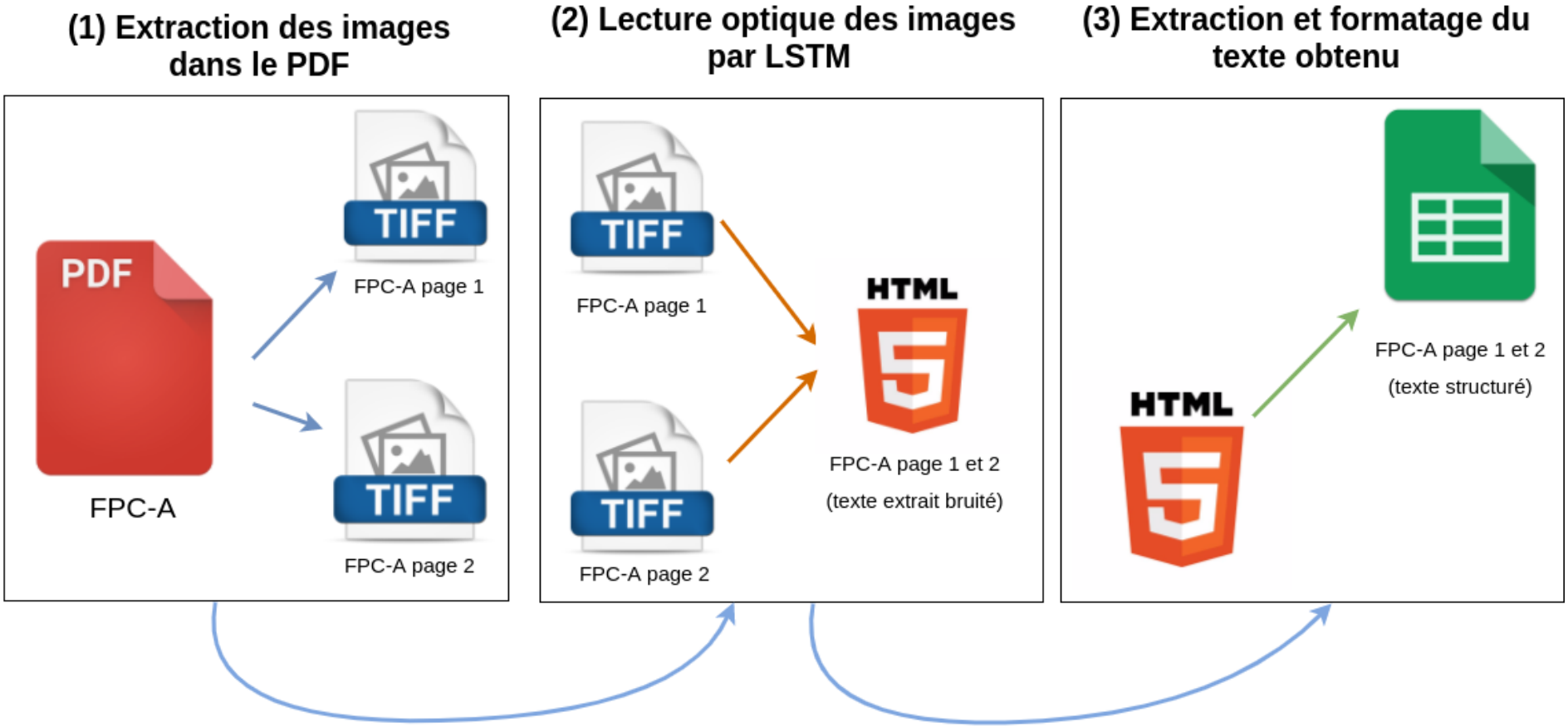
Librairies d'OCR : pourquoi Ocropy ?

- Breuel (2008) : The OCRopus open source OCR system.
- Breuel, Ul-Hasan, Mayce et Shafait (2013) : High performance of OCR for printed English and Fraktur using LSTM.
- Springmann et al (2014) : évaluation des performances de 3 OCRs (Ocropy : 81,66%, ABBYY FineReader : 80,57%, Tesseract : 78,77%)
- Drobac, Kauppinen et Lindén (2017) : Ocropy on historical Finnish texts (93% – 95,21)%)
- Outils libres de NVIDIA Research Projects [*github.com/NVlabs*]
- Le project *In Codice Ratio* (numérisation des archives du Vatican)

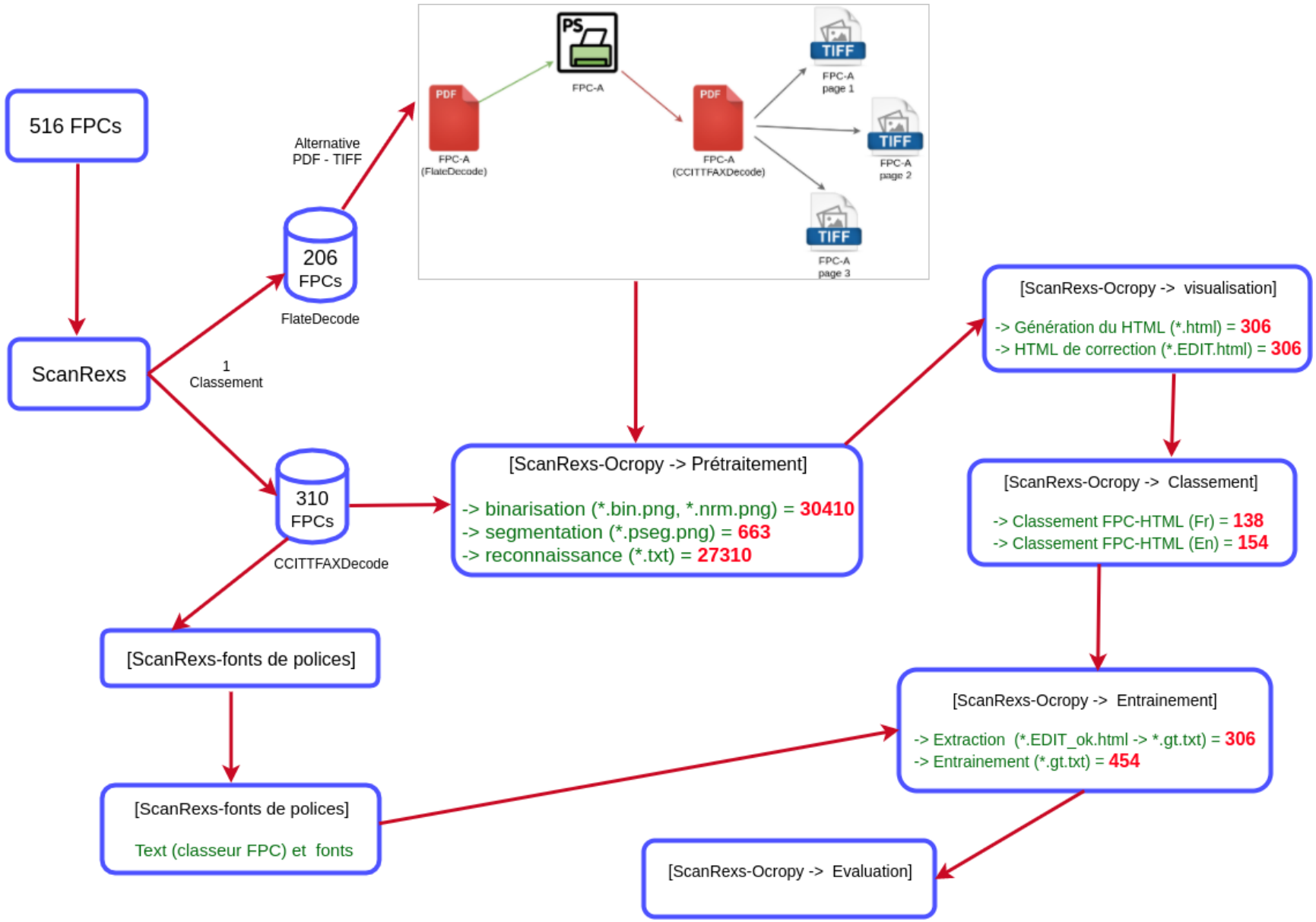
Plan

Extraire du contenu dans les FPC avec ScanRexs

Chaîne de traitement "ScanRexs"



Chaîne de traitement adoptée : Quelques chiffres



Plan

Application d'une méthodologie incrémentale

1. Génération des fichiers d'annotation à partir du modèle par défaut d'Ocropy
2. Annotation des FPC (correction manuelle)
3. Constitution d'un lot d'apprentissage/évaluation avec une quantité de données limitée
4. Apprentissage et évaluation des modèles obtenus
5. Sélection du meilleur modèle et incrémentation du nombre de données d'apprentissage
6. Mise à jour des codecs et paramètres de l'outil, puis apprentissage et évaluation de nouveaux modèles
7. ...

Plan

Correction des documents numérisés

- Correction

124 FPC (sur 516)
13k segments (lignes)

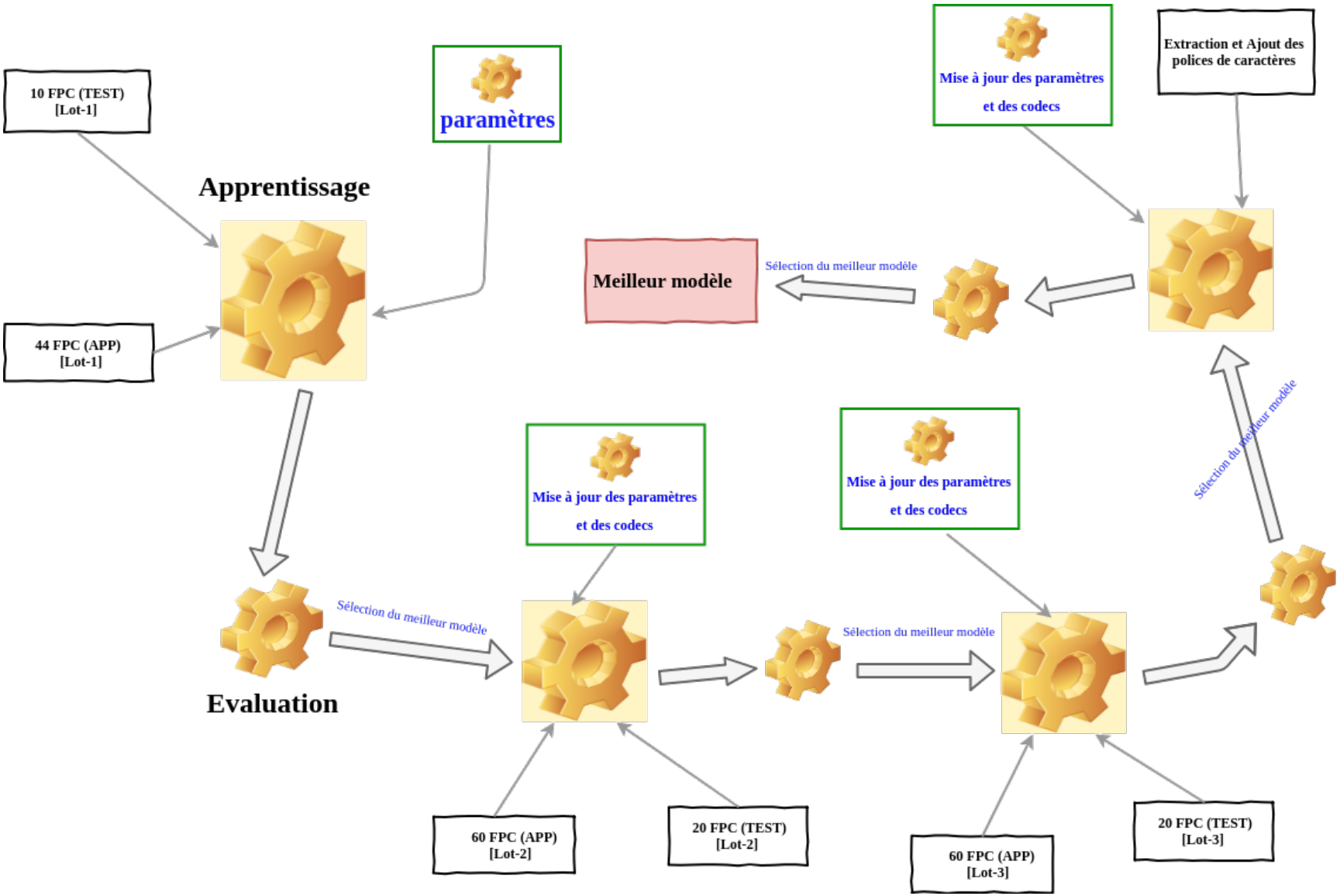
- Estimation du temps

Estimation temps de correction			
html	1-2 p.	1-5 p.	5 p. +
A	5mn	10mn	15mn+
B	10mn	20mn	30 mn+
C	15mn	25mn	35 mn+
D	15-20mn	25-40mn	60mn +

- Types d'erreurs observées
  - Minuscule/majuscule, caractères absents, écriture manuscrite
  - Erreurs d'OCR génériques : *l = 1 ; e = c*
  - Erreurs d'OCR du modèle de langue : *ü = é—è, __=i, 2 = é, a = à*

Plan

Chaîne de traitement adoptée : Méthode incrémentale



Récapitulatif des entraînements réalisés

- Apprentissage automatique : adaptation ocropy aux données
- Itération des modèles et calcul de performance (taux d'erreurs)
- Utilisation de fiches corrigées (*ground truth*)

nb de passes	18
données d'entraînement	44 à 109 FPC (10355 lignes)
données de test	10 à 15 FPC (1594 lignes)
nbre de modèles générés	1285 (980 + 305) modèles
nbre d'itérations atteint	plus de 340000
meilleur modèle obtenu	CER $4,241\%$ = CAR $95,8\%$
nb itérations meilleur modèle	290000

Paramètres optimisés pour l'apprentissage

- Paramètres de l'OCR
  - Modèles
    - Par défaut : anglais, 100K itérations initiales
    - Vierge
    - Meilleurs modèles précédents
    - Couches cachées (hiddensize) : 100 à 300
  - Nombre d'itérations (50k à 300k)
  - Segmentation et binarisation
    - vscale de 1.0 à 2.0
    - escale de 1.0 à 1.5
    - Meilleurs modèles précédents
  - Learning rate LSTM : 1e4 (au lieu de 1e5)

Évaluations selon les itérations I

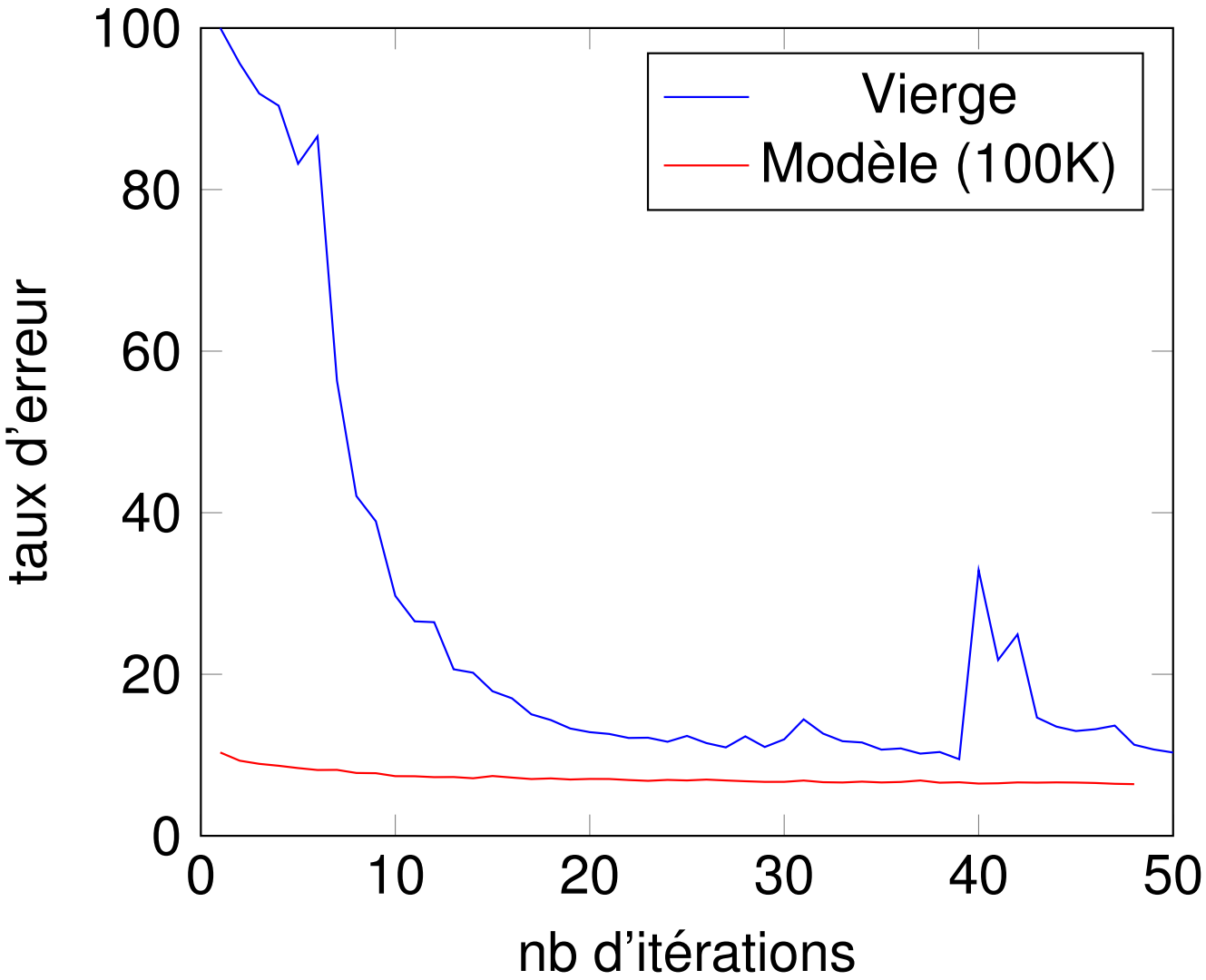


FIGURE – Amélioration de l'OCR avec modèle par défaut ou vierge

Évaluations selon les itérations II

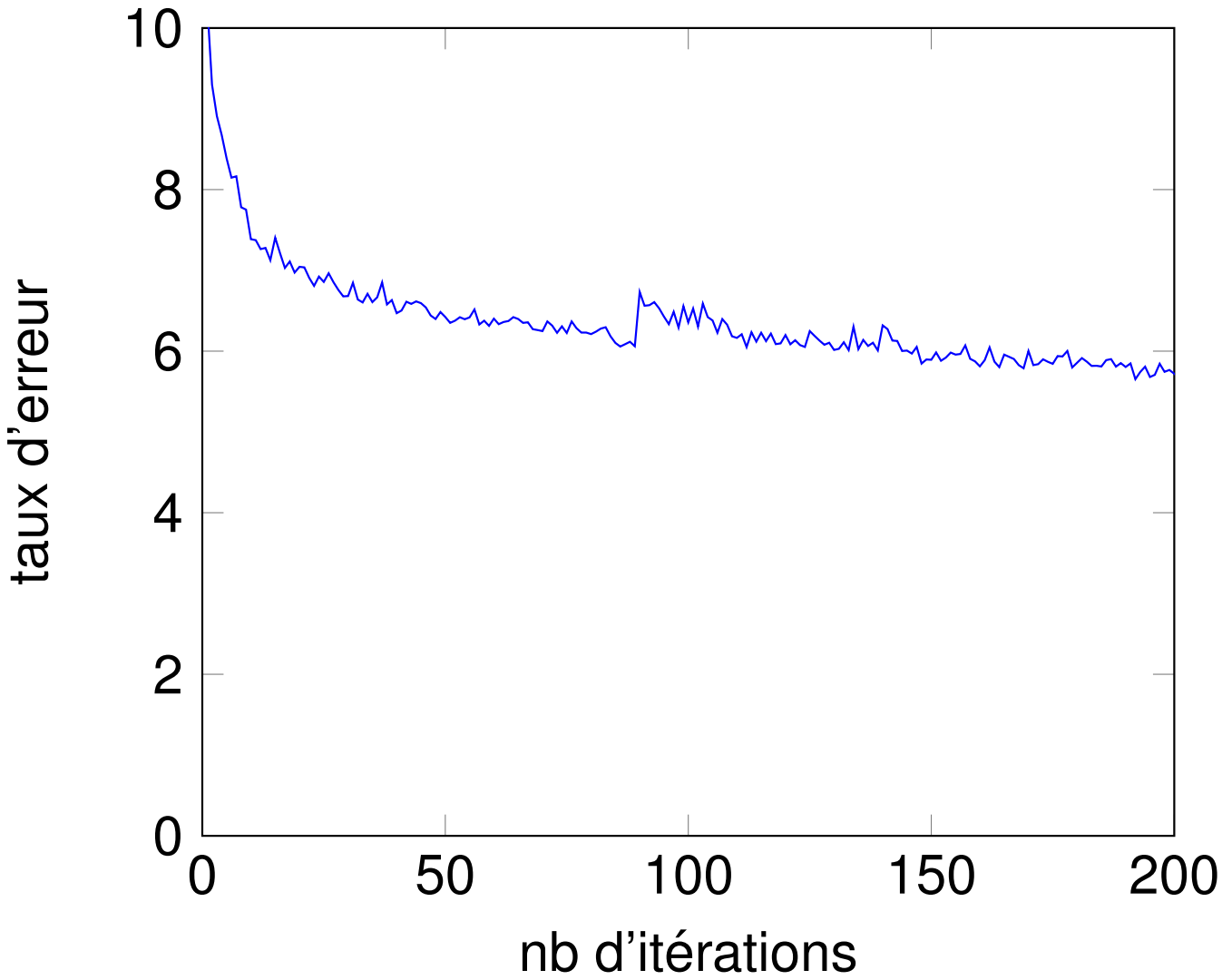
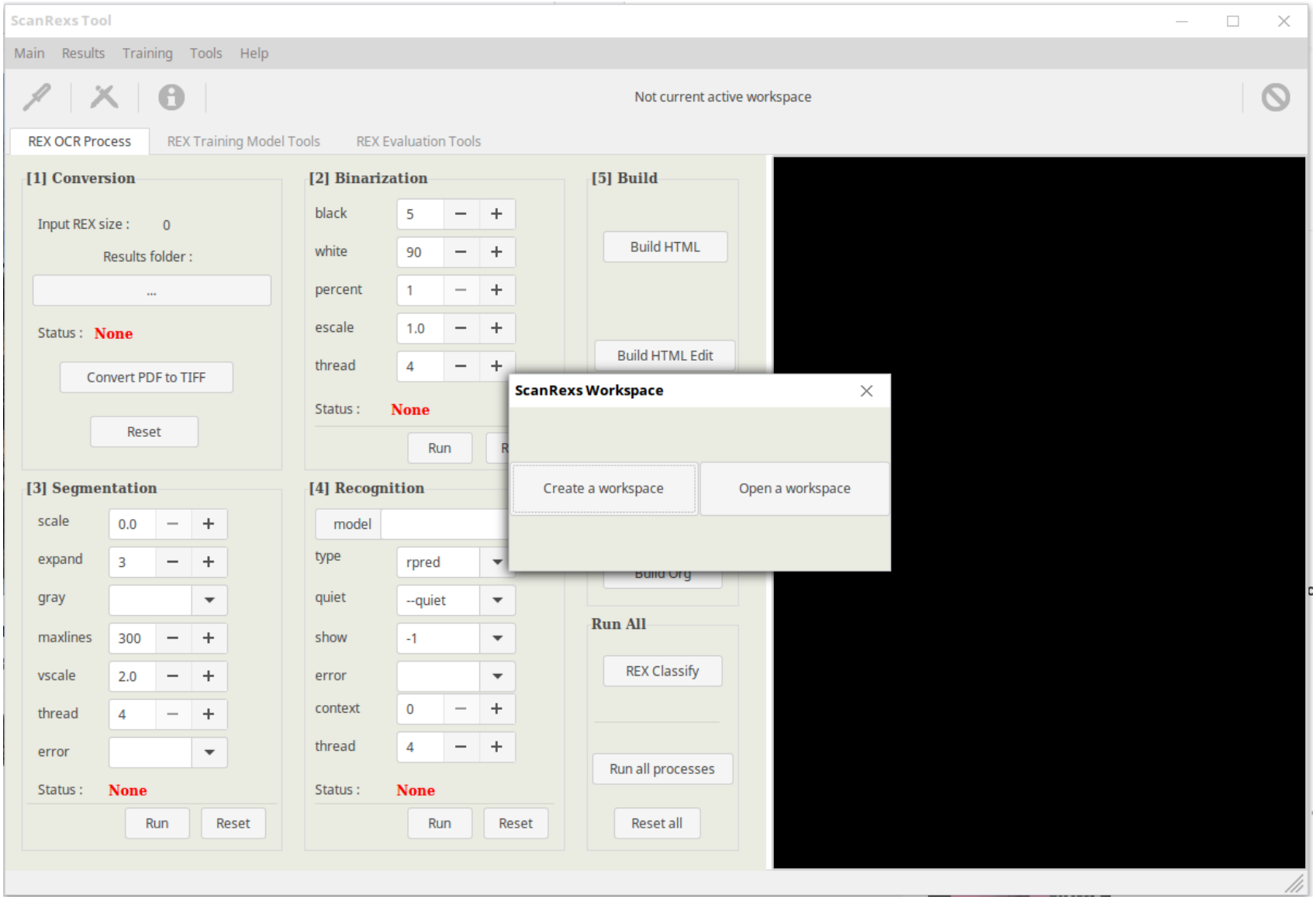


FIGURE – Amélioration du taux d'erreur de l'OCR selon le nombre d'itérations

Interprétation des résultats

- Quelques éléments d'interprétation
  - Le nombre d'itération améliore le WER
    50k : 10,3% $\rightarrow$ 300k : 4,2%
  - Moins marqué avec le modèle par défaut (9,7% $\rightarrow$ 6,3%)
  - Impact de notre travail sur la réduction du taux d'erreur
  - Impact de la quantité d'entraînement sur les performances
- Livraisons pour le projet
  - 62% des fiches numérisées
  - Logiciel ScanRexs
    - Intégration / adaptation d'OCRopy
    - Interface graphique
    - Cycle d'apprentissage / annotation / évaluation
  - Sous-corpus OCR annotée / corrigée manuellement
  - Analyses, rapports, etc.
- La suite
  - Utilisation / amélioration par le CNES
  - Expériences complémentaires . . .

Coup d'œil sur ScanRexs I



Coup d'œil sur ScanRexs II

1. ScanRexs s'organise en 3 processus
   - Processus de numérisation (OCR)
   - Processus d'entraînement des modèles (fonts/ground thruth)
   - Processus d'évaluation

2. Utilisation possible
   - Interface graphique
   - Ligne de commande (CLI)
   - Environnement virtuel (Vagrant)

3. Avantages
   - Première interface graphique de Ocropy
   - Chaîne de traitement automatisée basée sur le concept de *workspace*
   - Adaptable, multitâche, facile d'utilisation

Terminologie et ontologie

Objectifs

- Repérer la terminologie et la façon dont on peut construire une ontologie du / des risques appliquée au domaine de l'aérospatial
- Rechercher des relations (génériques, spécifiques, sémantiques) pertinentes pour le domaine, par exemple des relations causales

Terminologie

- Sigles et acronymes (sigles systèmes, acronymes industriels...)
- Ensemble sémantique par domaine. Par exemple "**pression**" : unité de mesure (bar), hydraulique, kPa, MPa, thermodynamique, etc.
- Vocabulaire de la description du risque (ouvert, clos, observabilité, possible)

Extraction terminologique

- TermSuite : mots simples et expressions poly-lexicales, collocations (pipeline par défaut). Sortie : fichier .tsv + tri par catégorie grammaticale

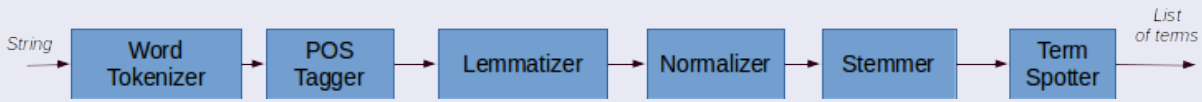


FIGURE – Pipeline de TermSuite

- Unitex / GramLab : graphes pour la détection des constructions verbales exprimant la causalité ; constitution de lexiques et dictionnaires électroniques contenant du vocabulaire spécialisé

Spécificités textuelles

Extraction du vocabulaire spécifique

- Utilisation du logiciel de textométrie TXM
- Spécificités
  - **Positives** : mots sur-représentés dans la partie
  - **Négatives** : mots sous-représentés dans la partie
- Parties selon
  - La classe de gravité
  - Par intervalle de temps

Classification des fiches en gravité

Du texte à la gravité

- Apprentissage automatique à partir d'un tableur contenant la transcription manuelle de 228 fiches (166 FR ; 62 EN)
- Utilisation du texte en français (tokenisation, sélection du vocabulaire)
- Prédiction de la classe de gravité : G0A, G0B et G123 (regroupés)
- Expérience de classification dans Weka et choix du meilleur classifieur

Prédiction de la gravité

CLASSIFICATION MODEL	OCC	BOOL
J48 pruned crossval	58%	59%
J48 unpruned crossval	56%	59%
NaiveBayes default cross-val 10	56%	59.7%
MultinomialNB default cross-val 10	54%	57%
SMO default cross-val 10	**62.8%**	**66%**
RF default cross-val 10	60%	62%
CLUSTERING MODEL	**OCC**	**BOOL**
SimpleKMeans numclust.3 s-10 class2clust	53%	51%
SimpleKMeans numclust.3 s-100 class2clust	**55%**	**55%**
EM numclust.3 s-10	52%	45%
EM numclust.3 s-100	51%	43%

TABLE – Résultats des expériences sur corpus-FPC-FR sans fiche dupliquée représentation intermédiaire (stoplist et hapax) en nombre d'occurrences et booléens

Quelques conclusions en cours de projet

- Numérisation des fiches FPC
  - Développement et adaptation d'un OCR
  - Annotation de fiches pour son entraînement
  - Évaluations pour valider sa fiabilité . . .mais à confirmer !
  - Livraison du logiciel et des données en cours
- Premiers travaux en signaux faibles
  - Difficulté de détecter automatiquement la gravité
  - Extraction de vocabulaire selon
    - La gravité de la fiche (risque)
    - La date de la fiche (campagnes)
  - Extraction et clustering de relations causales