



Atelier Outils de numérisation de documents: Transcription, OCR, HTR...

Motasem ALRAHABI, ObTIC - Scai, Sorbonne Université

28/10/2021

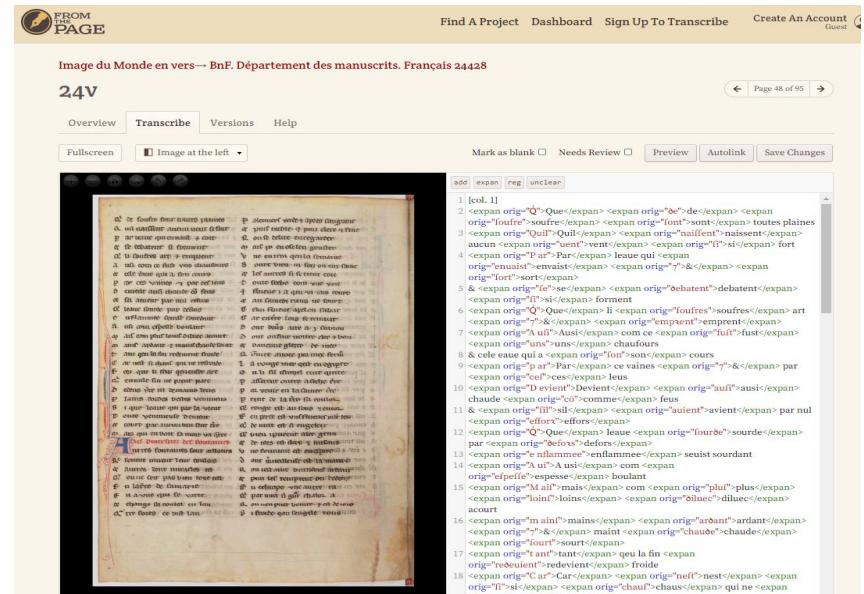
Plan

- Archives
- Transcription
- Numérisation de documents
- Prise en main de quelques outils: Abbyy, Kraken et Transkribus

Transcription

Transcription

- Il existe des outils pour aider à transcrire et annoter les images scannées.
 - Ils proposent en vis-à-vis de l'image, un bloc texte généralement accompagné de fonctionnalités de mise en forme, de structuration et d'enrichissement.
 - Exemples: [From The Page](#)
 - [Scribe](#), [Transcrire](#), [T-Pen](#)...
- Plugins: [e-man](#), [Scripto](#)
- Certains de ces outils permettent aussi le travail collaboratif: [Recital](#), [Transcrire](#), [Tact](#), [Transcribathon](#)...



Numérisation

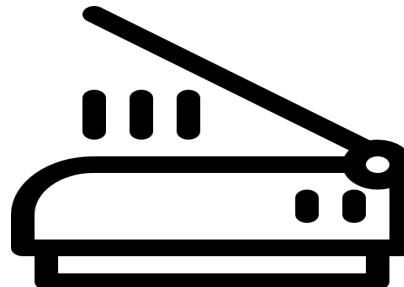
Numérisation et transformation digitale

- La numérisation de documents facilite l'analyse des contenus textuels ainsi qu'une meilleure qualité de lecture et de stockage.
 - Numérisation des fonds d'archives publiques, numérisation des fonds des bibliothèques, numérisation des documents d'entreprise (notes de frais, formulaires papier, factures, chèques, reçus...), reconnaissance automatique des plaques d'immatriculation, reconnaissance d'écriture manuscrite en temps réel (écran tactile...), dossiers hospitaliers, etc.



Numérisation

- La numérisation d'un texte s'effectue en trois phases distinctes :
 - Scanner les pages, reconnaître les chaînes de caractères et améliorer la qualité de la sortie.
- Photographie (scan) des pages:
 - La qualité de la photographie (jpeg, pdf...) a un impact sur l'étape suivante. Il est conseillé d'effectuer des images avec une résolution de 300 dpi (dots per inch).



Numérisation

- Océrisation: reconnaissance optique de la photographie de la page
 - Procédé informatique pour la transformation d'images de textes manuscrits (Handwritten Text Recognition), d'images imprimés ou dactylographiés (Optical Character Recognition) en fichiers de texte (texte brut, xml comme [Alto...](#)).
- Etapes:
 - Prétraitement : réalignement, corrections de contraste, binarisation (bicolore), détection de contours, suppression du bruit (déparasitage)...
 - Segmentation en lignes, en mots et en caractères, détection des blocs et des zones...
 - Reconnaissance proprement dite des caractères: comparaison avec une bibliothèque de formes connues et sélectionner les formes les plus proches, selon une distance ou une vraisemblance.



Numérisation

- Principe:
 - Comparer la forme à reconnaître avec une base de données de formes.
 - Le système choisit ensuite la forme la plus proche.
 - Techniques utilisées: apprentissage profond (réseaux de neurones), méthodes métriques (distance Levenshtein...), méthodes probabilistes (chaînes de Markov...)



Numérisation

- Correction des erreurs: l'OCR ne permet pas toujours d'obtenir un résultat parfait.
- Utilisation de méthodes linguistiques et contextuelles pour réduire le nombre d'erreurs:
 - Systèmes à base de règles, méthodes statistiques basées sur des dictionnaires de mots, de syllabes, de N-grammes, éliminer des solutions incorrectes, etc.
- Exemples d'erreurs pour le français:

u / n
a / o
1 / 1
! / 1
ponctuation
accentuation
...

fflitzung und Benutzung von Miquels Buch sei



Vgl. dazu noch § 29). Die Annahme von r^6

Numérisation

- Les techniques d'OCR sont en progrès constant pour répondre à une demande très forte, mais la qualité de segmentation et de reconnaissance dépend de facteurs liés au document original et à sa numérisation.
- Difficultés pour l'OCR:
 - Différents styles typographiques (gras, italique, normal...) et de polices.
 - Documents historiques, anciennes graphies, noms obsolètes...
 - Diversité des langues, nombre variable du vocabulaire...
 - Défauts d'impression (caractères empâtés, bavures...)
 - Documents en colonnes ou illustrés, lecture non linéaire...
 - Polices (trop) petites ou grandes...
 - Résolution de l'image scannée, contraste et luminosité, qualité du support...
 - ...



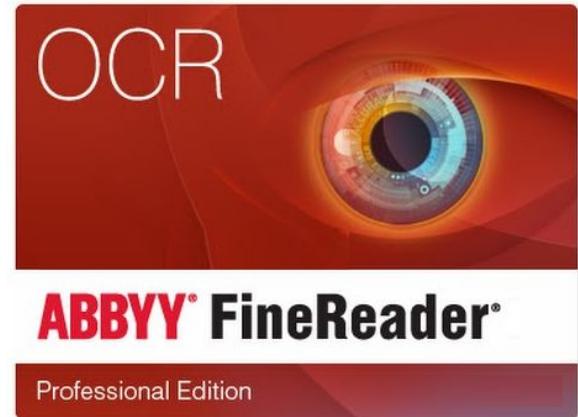
Numérisation

- Exemples d'outils:
 - [Tesseract](#)
 - [Kraken](#)
 - [eScriptorium](#)
 - [Transkribus](#)
 - [Abbyy FineReader](#)
 - [Ocr4All](#)
 - Google Cloud Platform : [API Vision OCR](#)
 - Microsoft Azure Cognitive Services : [Vision par ordinateur OCR](#)
 - Amazon Web Services : [Amazon Textract](#)
 - [OCR Space](#)
 - ...

Abbyy FineReader

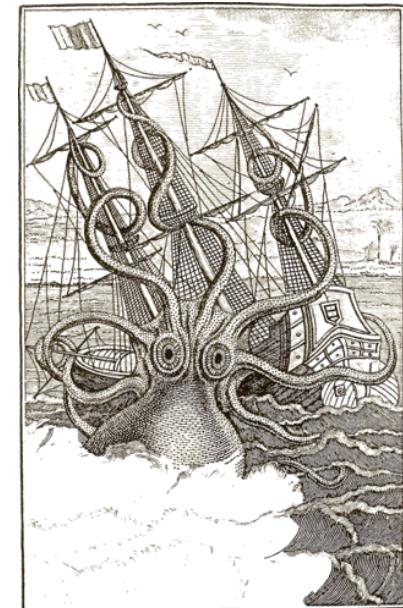
- Abbyy FineReader est un logiciel d'OCR propriétaire.
- Le processus se décompose en 3 étapes :
 - Ouvrir (numériser) le document, le reconnaître puis le sauvegarder dans un format courant (DOC, RTF, XLS, PDF, HTML, TXT, etc.) ou exporter les données directement vers une application de Microsoft Office telle que Microsoft Word, Excel ou Adobe Acrobat.

- Démonstration.



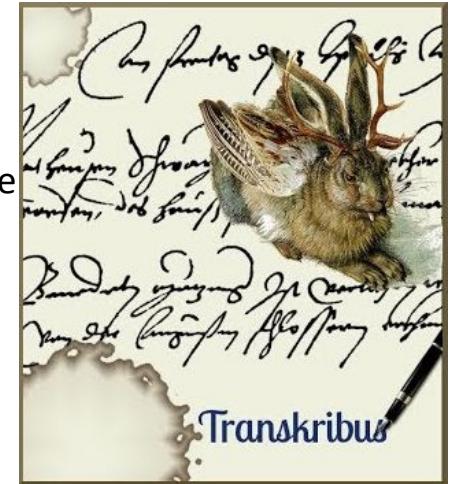
Kraken

- Développé en python (Kiessling, 2019), Kraken est un logiciel open source d'OCR et HTR qui fonctionne grâce à des réseaux de neurones récurrents.
- Kraken est en ligne de commande mais avec eScriptorium, il est en train de se doter d'une interface graphique.
- Il permet de binariser et de segmenter une page, de la transcrire (d'abord à la main pour créer les données d'entraînement, sinon automatiquement), d'effectuer l'entraînement d'un modèle d'OCR/HTR et la reconnaissance d'écriture.
- Export du résultat en texte brut ou au format XML ALTO.



Transkribus

- Transkribus est un logiciel avec interface graphique (GUI) qui permet de segmenter une page, de la transcrire (à la main ou automatiquement) et de l'annoter avant de l'exporter dans plusieurs formats possibles.
 - On doit d'abord transcrire manuellement une centaine de pages, afin de permettre d'entraîner un modèle de transcription. Le logiciel va alors reconnaître automatiquement le texte des pages suivantes et en proposer une transcription ligne à ligne.
- L'interface de Transkribus est open source, mais ce n'est pas le cas de son système d'entraînement pour la transcription automatique : les modèles de transcription ne sont donc accessible que par l'intermédiaire de Transkribus.



e-Scriptorium

- Scriptorium est une interface web pour le traitement d'images contenant du texte, leur segmentation, leur transcription et leur annotation. Actuellement développé comme une couche graphique pour Kraken, cette interface a vise à être compatible avec plusieurs solutions de transcription automatique. eScriptorium est développée dans le cadre du projet SCRIPTA.
- Si la majeure partie des développements actuels se passent sur eScriptorium, qui intègre la brique Kraken, une bonne partie des expérimentations de transcriptions de la phase 1 se sont déroulés sur Transkribus.

Partie pratique

Ljudmila Petković, ObTIC - Scai, Sorbonne Université

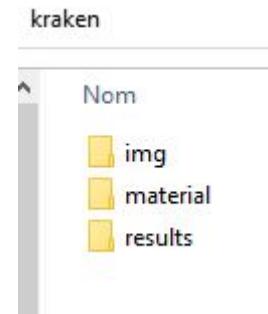
28/10/2021

Kraken¹

¹ Le contenu des diapositives consacrées à Kraken s'appuie sur le dépôt GitHub Simon Gabay, *Distant Reading I: hacker les humanités*, Genève: Université de Genève, 2020, https://github.com/gabays/DistRead_1/blob/master/DistRead_1_4/DistRead_1_4.ipynb (consulté le 27 février 2022).

Préliminaires (en ligne de commande, Mac ou Linux)

1. Installer kraken `pip install kraken`
2. Basculer en mode `python`, ensuite importer la librairie `import kraken`
3. Télécharger un [modèle \(Gabay et al., 2020\)](#)
4. Déplacer le modèle dans le dossier `material` que nous créons
5. Déposer une [image](#) dans le dossier `img` que nous créons
6. Nous créons un dossier `results` où les transcriptions seront stockées



Premier test

- Lancement de la chaîne de traitement sur l'image

Arguments :

- `-i` : fichier d'entrée, *input*
- Chemin vers l'image à numériser : `10.jpg`
- Chemin vers le fichier qui va être généré lors de la numérisation : `ocr_result.txt`
- Binarisation : `binarize`
- Segmentation : `segment`
- OCRisation : `ocr`
- `-m` : modèle
- Chemin vers le modèle d'OCR : `OCR17.mlmodel`

```
kraken -i ./img/10.jpg ./results/ocr_result.txt binarize segment ocr -m ./material/OCR17.mlmodel
```

- Afficher le résultat de la transcription : `cat ./results/ocr_result.txt`

Transcription

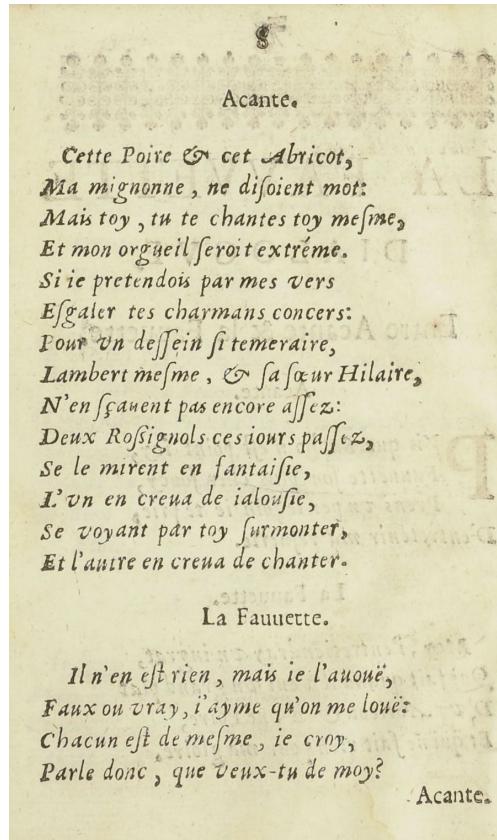
5

Acante.

Cette Poire & cet Abricot,
Ma mignonne, ne diſoient mot:
Mais toy, tu te chantes toy meſme,
Et mon orgueil ſeroit extréme.
Si ie pretendois par mes vers
Eſgaler tes charmans concers
Four vn deſſein ſi temeraire,
Lambert meſme, & ſa ſœur Hilaire,
N'en ſçauent pas encore aſſez:
Deux Rosignols ces iours paſſe,
A C1
Se le mirent en fantaſie,
L'vn en creua de ialouſie,
Se voyant par toy ſurmonter,
Et l'autre en creua de chanter-
La Fauvette.
Il n'en eſt rien, mais ie l'auoue,
Faux ou vray, i'ye qu'on me loue:
Chacun eſt de meſme, ie croy,
Parle donc, que eux-tu de moy

7

Acante.



La chaîne de traitement décomposée

- Binarisation : kraken -I ./img/10.jpg -o .png binarize

Acante.

*Cette Poire & cet Abricot,
Ma mignonne, ne disoient mot:
Mais toy, tu te chantes toy mesme,
Et mon orgueil seroit extréme.
Si ie pretendois par mes vers
Esgaler tes charmans concers:
Pour v'n dessein si temeraire,
Lambert mesme, & sa sœur Hilaire,
N'en ffauent pas encore affez:
Deux Rosignols ces iours passiez,
Se le mirent en fantaisie,
Ivn en creua de jalouzie,
Se voyant par toy surmonter,
Et l'autre en creua de chanter.*

La Fauvette.

*Il n'en est rien, mais ie l'auouë,
Faux ou vray, l'ayme qu'on me louë:
Chacun est de mesme, ie croy,
Parle donc, que veux-tu de moy?*

Acante.

La chaîne de traitement décomposée

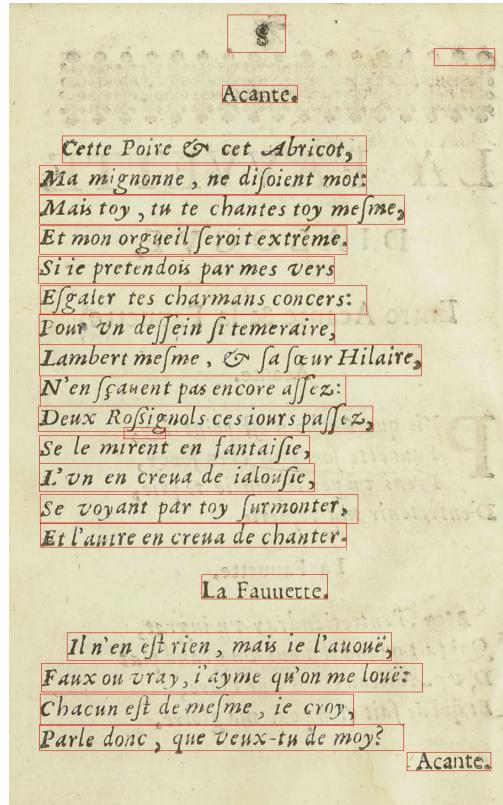
- Binarisation + segmentation : `kraken -I ./img/10.jpg -o .json binarize segment`
- Afficher le résultat de la segmentation : `cat ./img/10.json`

```
{"text_direction": "horizontal-lr", "boxes": [[396, 22, 501, 93], [386, 155, 522, 188], [96, 251, 647, 302], [55, 307, 647, 362], [55, 363, 716, 418], [55, 421, 612, 475], [53, 479, 589, 530], [56, 534, 648, 588], [54, 589, 590, 646], [55, 646, 747, 705], [56, 705, 610, 762], [53, 762, 659, 810], [208, 802, 283, 824], [55, 818, 544, 871], [55, 871, 553, 924], [56, 929, 632, 984], [57, 983, 611, 1029], [349, 1081, 577, 1127], [105, 1191, 693, 1243], [58, 1249, 749, 1303], [56, 1307, 624, 1366], [55, 1364, 715, 1414], [770, 86, 880, 118], [721, 1417, 873, 1452]], "script_detection": false}
```

- série de rectangles, définis par leurs quatre coins

Segmentation « en pratique »

- Division du texte en lignes
- La transcription se fera ensuite à partir de chaque ligne



OCRisation d'un échantillon

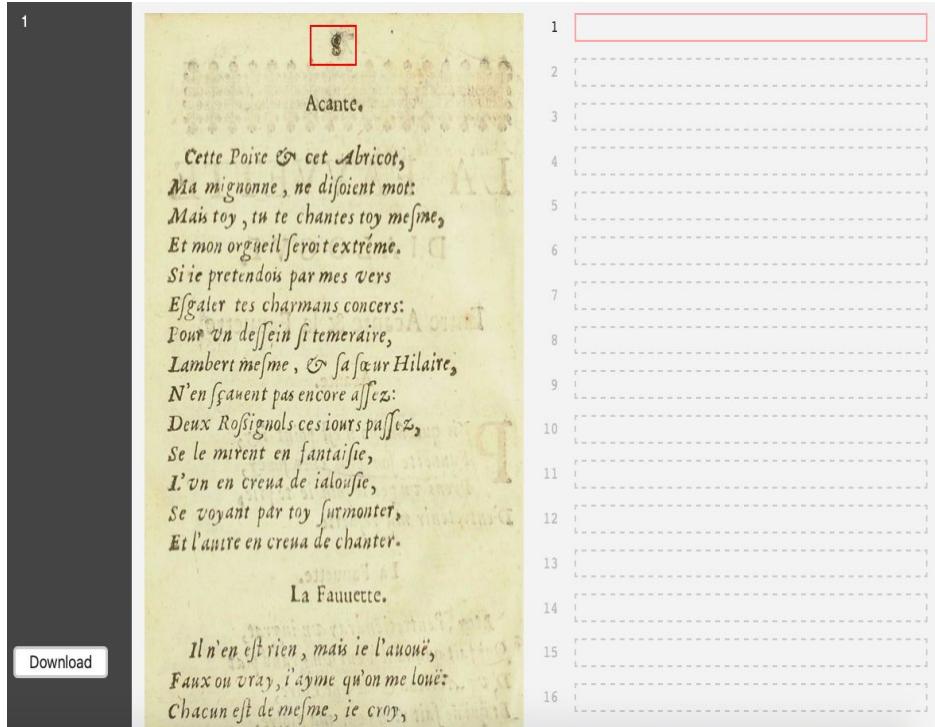
- Télécharger deux nouvelles images ([image 2](#), [image 3](#))
- Effacer l'image numérisée faite précédemment : `rm ./img/*png`
- Binariser toutes les images d'un seul coup : `kraken -I "./img/*.jpg" -o .png binarize`
- Créer un sous-dossier `bin` dans le dossier `img` où les images binarisées seront stockées : `mkdir ./img/bin; mv ./img/*png ./img/bin/`
- Binariser puis segmenter les images dans le fichier `img` : `kraken -I "img/*jpg" -o .json binarize segment`
- Déplacer les fichiers `.json` produits dans le sous-dossier `seg` : `mkdir ./img/seg; mv ./img/*json ./img/seg/`

Extrait du journal des opérations

```
WARNING:root:scikit-learn version 0.24.1 is not supported. Minimum required version: 0.17.  
Maximum required version: 0.19.2. Disabling scikit-learn conversion API.  
Loading ANN default ✓  
Binarizing ✓  
Segmenting ✓  
Processing [#####
] 100%  
Writing recognition results for ./img/Collectif1660_recueil_bpt6k853407j_12.jpg ✓  
Binarizing ✓  
Segmenting ✓  
Processing [#####
] 100%  
Writing recognition results for ./img/Collectif1660_recueil_bpt6k853407j_13.jpg ✓  
Binarizing ✓  
Segmenting ✓  
.....
```

Création des données d'entraînement

- Créer une interface de transcription sous forme d'un fichier html afin de pouvoir générer les données d'entraînement (vérité terrain, angl. *ground truth*) :
ketos transcribe
-o ./material/test.html
./img/10.jpg



Création des données d'entraînement

- À priori, il existe la possibilité du pré-remplissage automatique des transcriptions, avec le but d'économiser du temps nécessaire pour la transcription / relecture manuelle (à vérifier) : `ketos transcribe -o ./material/test_prefill.html --prefill ./material/OCR17.mlmodel ./img/10.jpg`
- Télécharger les données

The screenshot shows the Ketos application interface. On the left, a dark panel displays a "Download" button, which is circled in red. To the right, there is a document page with handwritten lyrics in French. The lyrics are partially transcribed with green boxes. A vertical list of numbered items on the right side corresponds to these green boxes, indicating the transcription status. The items are:

1	8
2	Acante.
3	Cette Poire & cet Abricot,
4	Ma mignonne, ne disoient mot:
5	Mais toy , tu te chantes toy meſme,
6	Et mon orgueil ſeroit extrême.
7	Si ie pretendois par mes vers
8	Eſgaler tes charmans concers:
9	Pour vn deſſein fi temeraire,
10	Lambert meſme , & fa ſœur Hilaire,
11	N'en ſçauent pas encore afiez:
12	Deux Roſſignols ces iours pafiez,
13	Se le mirent en fantaiſie,
14	Lvn en creua de ialouſie,
15	Se voyant par toy furmonter,
16	Et l'autre en crena de chanter-

The lyrics on the page include:

1. Acante.
2. Cette Poire & cet Abricot,
3. Ma mignonne, ne disoient mot:
4. Mais toy , tu te chantes toy meſme,
5. Et mon orgueil ſeroit extrême.
6. Si ie pretendois par mes vers
Eſgaler tes charmans concers:
7. Pour vn deſſein fi temeraire,
Lambert meſme , & fa ſœur Hilaire,
N'en ſçauent pas encore afiez:
8. Deux Roſſignols ces iours pafiez,
Se le mirent en fantaiſie,
9. Lvn en creua de ialouſie,
Se voyant par toy furmonter,
10. Et l'autre en crena de chanter-

La Fauvette.

Il n'en eſt rien , mais ie l'auoué,
Faux ou vray , l'ayme qu'on me loue:
Chacun eſt de meſme , ie croys,

Extraction des données

- Création du dossier `gt` qui contient des lignes des images associées avec leurs transcriptions

```
ketos extract --output gt/ ./material/test_prefill.html
```

```
Reading transcriptions [########################################] 100%
```

```
| -gt
  |   -image_1.jpg
  |   -transcription_1.txt
  |   -image_2.jpg
  |   -transcription_2.txt
  |
  |   ...
```

Extraction des données

- Image :

Ma mignonne , ne disoient mot:

- Transcription : `cat ./gt/000003.gt.txt`

Ma mignonne , ne disoient mot:

Préparation de l'entraînement

- Télécharger le script écrit par Jean-Baptiste Camps (ENC | PSL) pour la création automatique des jeux de données :
 - `train` : données pour entraîner le modèle
 - `val` : données pour évaluer la performance des modèles créés successivement lors de l'entraînement
 - `test` : données non vues lors de l'entraînement pour tester le meilleur modèle

```
curl  
https://raw.githubusercontent.com/gabays/DistRead_1/master/DistRead_1_4/  
material/randomise_data.py --output ./material/randomise_data.py
```

Préparation de l'entraînement

- Lancer l'entraînement : `python ./material/randomise_data.py ./gt/*.png`
--> Génération des trois fichiers : `train.txt`, `val.txt`, `test.txt`
- Chaque fichier contient les chemins vers les images distribuées aléatoirement :
p. ex. `./gt/000000.png`
`./gt/000004.png`
`...`

Entraînement du modèle

```
ketos train -u NFD -t ./train.txt -e ./val.txt
```

```
WARNING:root:scikit-learn version 0.24.1 is not supported. Minimum required version: 0.17. Maximum required
version: 0.19.2. Disabling scikit-learn conversion API.
Building training set [########################################] 21/21
Building validation set [########################################] 2/2
[22.2418] alphabet mismatch: chars in training set only: {'q', 'I', 'S', 'v', 'N', 'r', 'l', 'h', 'n', 'E', "", 'y', 'R', 'æ', 'x', 'L', 'H', 'p', '.', 'f', 'l', 'D', 'z', 'u', 's', ',', '8', 'F'} (not included in accuracy
test during training)
Initializing model ✓
[W ParallelNative.cpp:206] Warning: Cannot set number of intraop threads after parallel work has started or after
set_num_threads call when using native parallel backend (function set_num_threads)
stage 1/∞ [########################################] 21/21
Accuracy report (1) 0.0000 55 55
...
stage 5/∞ [########################################] 21/21
Accuracy report (5) 0.0364 55 53
...
Moving best model model_1.mlmodel (0.0) to model_best.mlmodel
```

Le système choisit le modèle le plus performant

-u NFD : pour normaliser les caractères en Unicode

Entraînement du modèle

- Sauvegarder le meilleur modèle

```
cp ./model_best.mlmodel ./material/monModele.model
```

- Effacer les autres

```
rm -f ./*.mlmodel
```

Réglage fin du modèle existant

- Créer une sauvegarde (*back-up*)

```
cp ./material/OCR17.mlmodel ./material/OCR17.mlmodel.bk
```

- Entraîner le modèle

```
ketos train -i ./material/OCR17.mlmodel --resize add ./gt/*.png
```

- Sauvegarder le nouveau modèle

```
cp ./model_best.mlmodel ./material/monModeleFineTune.model rm -f /*.mlmodel
```

Réglage fin du modèle existant

```
WARNING:root:scikit-learn version 0.24.1 is not supported. Minimum required version: 0.17.
Maximum required version: 0.19.2. Disabling scikit-learn conversion API.
Loading existing model from ./material/OCR17.mlmodel ✓
Building training set [#####
] 21/21
Building validation set [#####
] 3/3 [8.9948] alphabet
mismatch: chars in training set only: {'I', '.', '-', 'é', 'b', 'H', '8', 'æ', ',', 'f', 'q',
'l', 'D', 'F', 'x', 'R', 'M', 'C', 'y', 'A', 'L', 'g', 'S', '&', 'h', 'E'} (not included in
accuracy test during training)
[8.9950] alphabet mismatch: chars in validation set only: {'ñ', 'ç', 'N'} (not trained)
[8.9962] Neural network has been trained on mode 1 images, training set contains mode <built-in
method mode of Tensor object at 0x7fbdcc8a2940> data. Consider setting `force_binarization`
stage 1/∞ [#####
] 21/21
Accuracy report (1) 0.8814 59 7
...
stage 9/∞ [#####
] 21/21
Accuracy report (9) 0.8644 59 8
Moving best model model_4.mlmodel (0.8983050847457628) to model_best.mlmodel
```

Évaluation du modèle

```
kertos test -m ./material/monModeleFineTune.model -e ./test.txt >./eval_model.txt
```

- Afficher le résultat de l'évaluation :

```
cat eval_model.txt
```

```
Loading model ./material/monModeleFineTune.model ✓
Evaluating ./material/monModeleFineTune.model
==== report ===

    7 Characters
    0 Errors
    100.00% Accuracy

    0 Insertions
    0 Deletions
    0 Substitutions

    Count      Missed      %Right
    6 0        100.00%     Latin
    1 0        100.00%     Common

    Errors   Correct-Generated

    Average accuracy: 100.00%, (stddev: 0.00)
```

OCRiser avec le nouveau modèle

```
kraken -i ./img/10.jpg ./results/ocr_result_nouveauModele.txt binarize  
segment ocr -m ./material/monModeleFineTune.model
```

```
WARNING:root:scikit-learn version 0.24.1 is not supported. Minimum required  
version: 0.17. Maximum required version: 0.19.2. Disabling scikit-learn  
conversion API.
```

```
Loading ANN default ✓
```

```
Binarizing ✓
```

```
Segmenting ✓
```

```
Processing [########################################] 100%
```

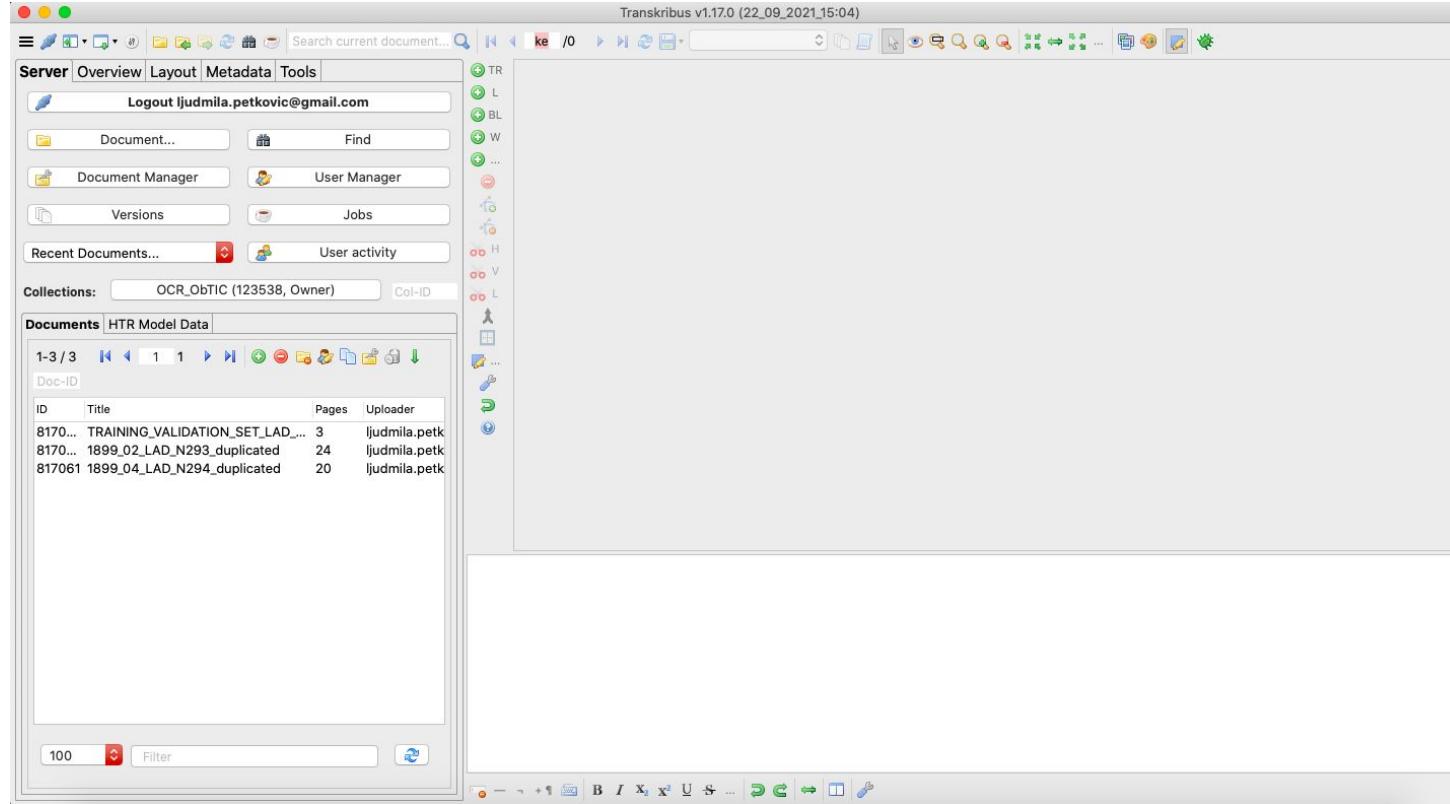
```
Writing recognition results for ./img/10.jpg ✓
```

Transkribus

Installation

- Créer le compte (<https://readcoop.eu>)
- Télécharger le logiciel
- Tutoriels
 - [Site](#)
 - [Vidéos Youtube](#)

Interface graphique



Importation des documents

- Créer sa propre collection
- Importer les documents et choisir le format

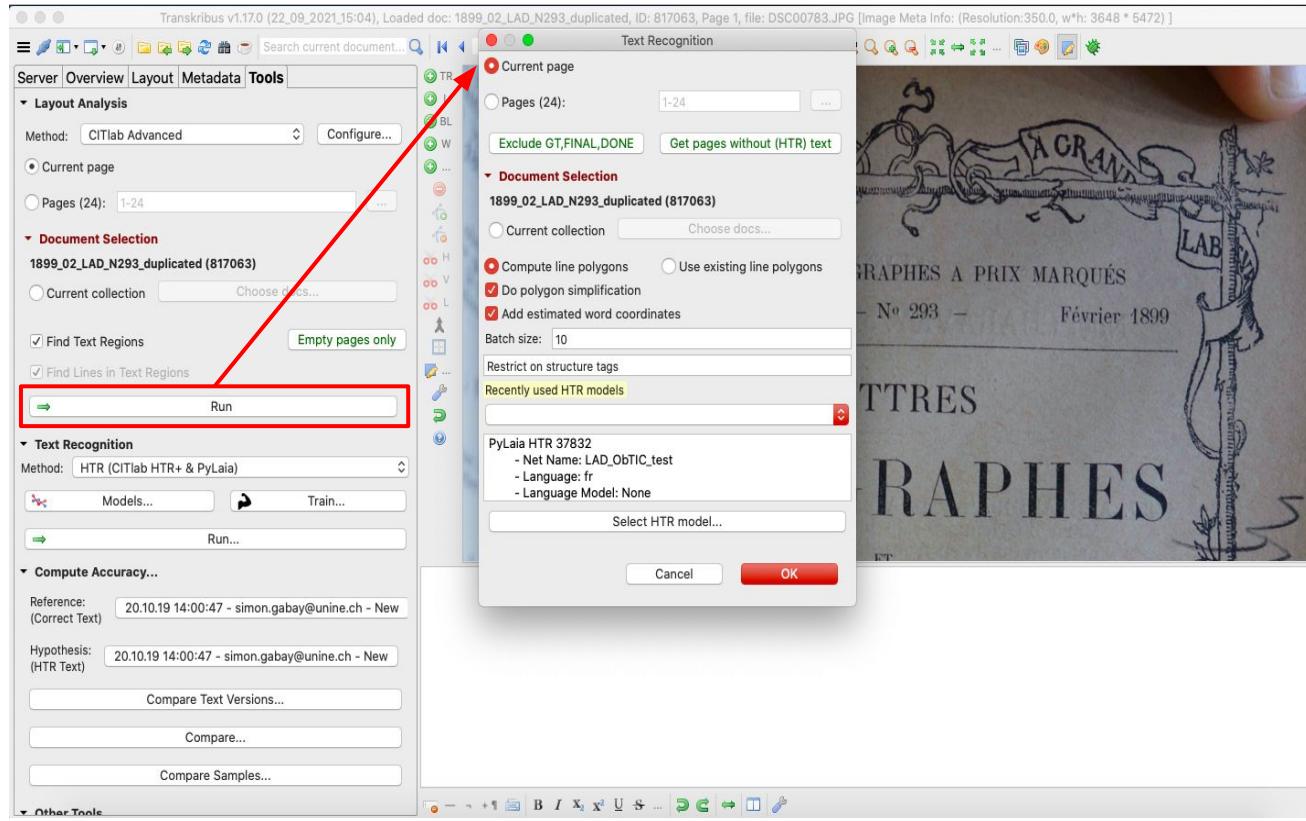
The screenshot shows a software interface for managing digital documents. At the top, there's a navigation bar with tabs: Server, Overview, Layout, Metadata, and Tools. A user session is shown as 'Logout ljudmila.petkovic@gmail.com'. Below the navigation bar are several buttons: Document..., Find, Document Manager, User Manager, Versions, Jobs, Recent Documents..., and User activity. A red box highlights the 'Collections:' dropdown menu, which contains the entry 'OCR_ObTIC (123538, Owner)'. The main area is titled 'Documents' and displays a table of document entries. The table has columns: ID, Title, Pages, and Uploader. The data includes:

ID	Title	Pages	Uploader
8170...	TRAINING_VALIDATION_SET_LAD...	3	ljudmila.petk...
8170...	1899_02_LAD_N293_duplicated	24	ljudmila.petk...
817061	1899_04_LAD_N294_duplicated	20	ljudmila.petk...

At the bottom, there are filters for '100' items and a 'Filter' button.

The screenshot shows a 'Document ingest / upload' dialog box. At the top, there are four radio button options: 'Upload via private FTP (also PDF files)', 'Upload single document' (which is selected and highlighted with a red circle), 'Upload via URL of DFG Viewer METS', and 'Upload via URL of IIIF manifest'. Below these are two input fields: 'Local folder:' and 'Title on server:', both of which have red boxes around them. At the bottom, there are buttons for 'Cancel' and 'Upload'.

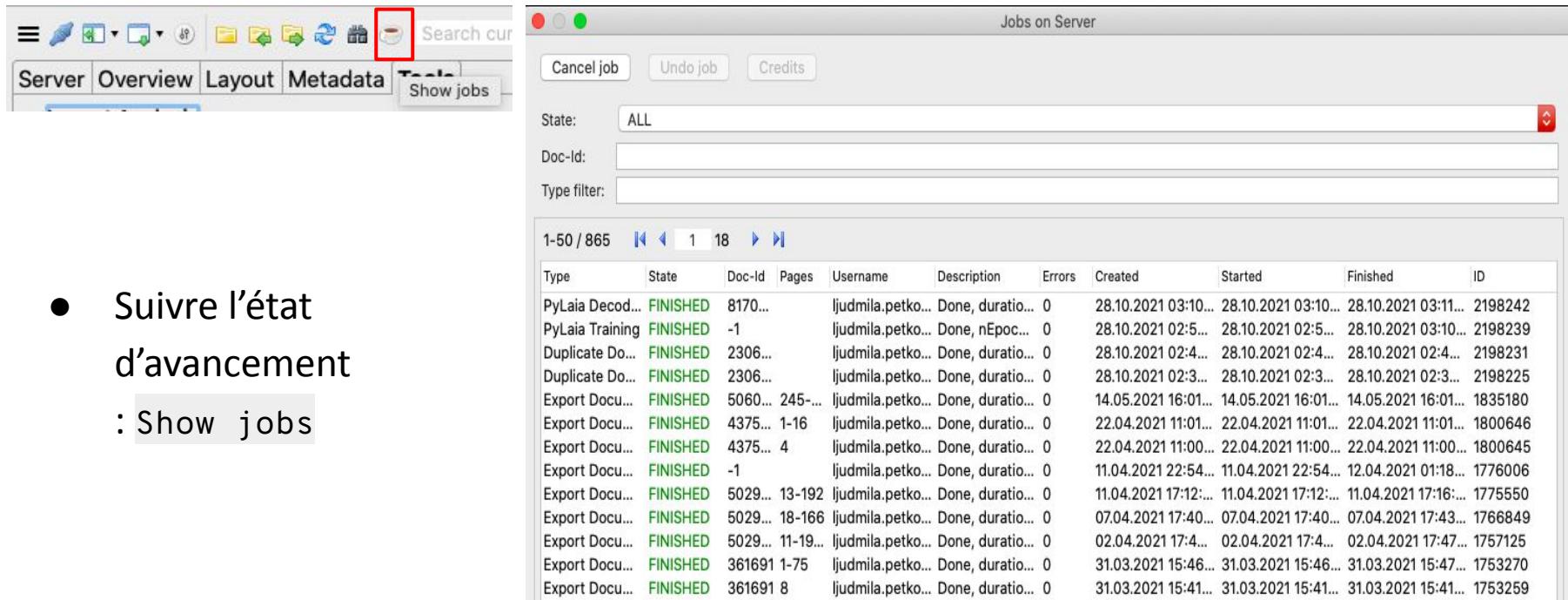
Transcription



Spécifier :

- les pages
- le modèle HTR / OCR
 - HTR (CITlab)
 - HTR + & PyLaia
 - Transkribus
 - OCR
 - (Block-segmentation + HTR + print)

Transcription

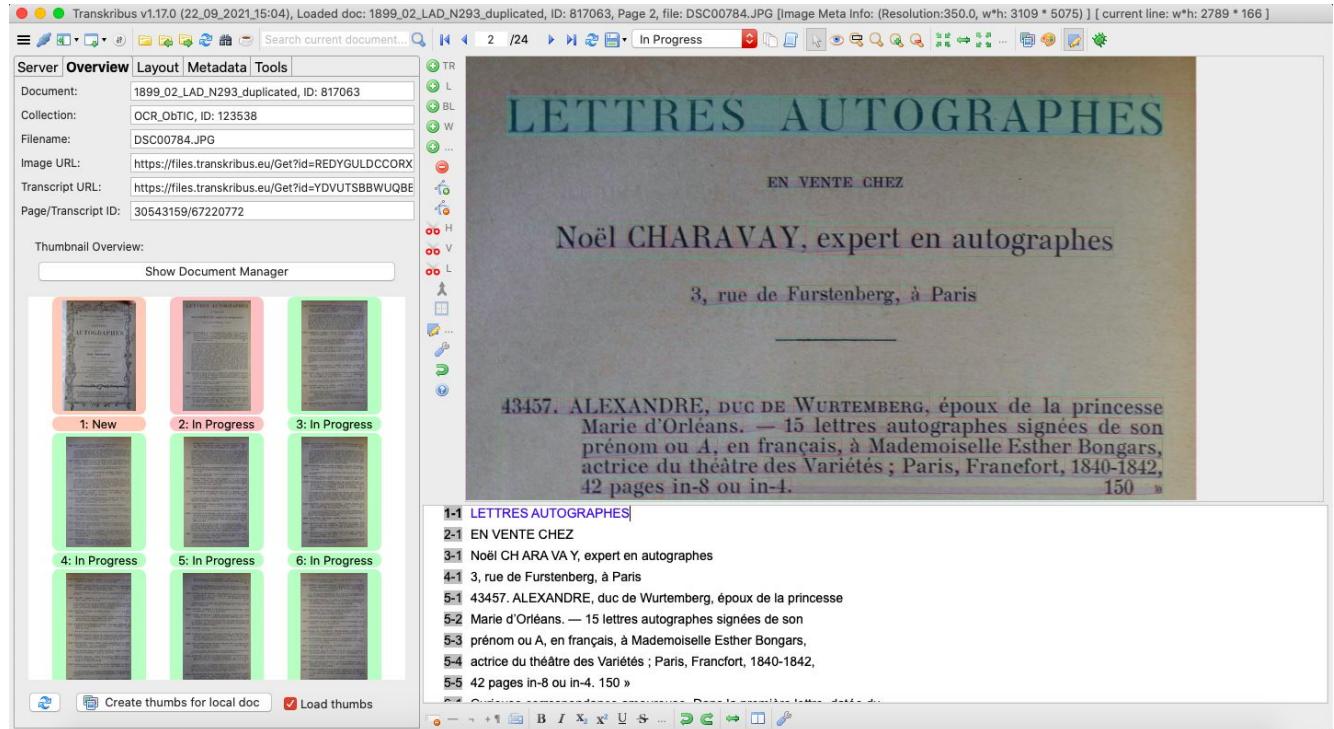


The screenshot shows a software interface titled "Jobs on Server". The toolbar at the top includes various icons and a search bar labeled "Search cur...". Below the toolbar, there are tabs: "Server", "Overview", "Layout", "Metadata", "Tools", and "Show jobs", with "Show jobs" being the active tab. The main area displays a table of transcription jobs. The table has columns: Type, State, Doc-Id, Pages, Username, Description, Errors, Created, Started, Finished, and ID. The "State" column shows mostly "FINISHED" status. The "Type" column includes entries like "PyLaia Decod...", "PyLaia Training", "Duplicate Do...", "Duplicate Do...", "Export Docu...", and "Export Docu...". The "Doc-Id" column contains values such as "8170...", "-1", "2306...", "2306...", "5060...", "4375... 1-16", "4375... 4", "-1", "5029... 13-192", "5029... 18-166", "5029... 11-19...", "361691 1-75", and "361691 8". The "Created" column shows dates ranging from 28.10.2021 to 31.03.2021. The "Started" and "Finished" columns show times corresponding to the creation dates. The "ID" column lists job identifiers from 2198242 to 1753270.

Type	State	Doc-Id	Pages	Username	Description	Errors	Created	Started	Finished	ID
PyLaia Decod...	FINISHED	8170...		ljudmila.petko...	Done, duratio...	0	28.10.2021 03:10...	28.10.2021 03:10...	28.10.2021 03:11...	2198242
PyLaia Training	FINISHED	-1		ljudmila.petko...	Done, nEpoch...	0	28.10.2021 02:5...	28.10.2021 02:5...	28.10.2021 03:10...	2198239
Duplicate Do...	FINISHED	2306...		ljudmila.petko...	Done, duratio...	0	28.10.2021 02:4...	28.10.2021 02:4...	28.10.2021 02:4...	2198231
Duplicate Do...	FINISHED	2306...		ljudmila.petko...	Done, duratio...	0	28.10.2021 02:3...	28.10.2021 02:3...	28.10.2021 02:3...	2198225
Export Docu...	FINISHED	5060...	245-...	ljudmila.petko...	Done, duratio...	0	14.05.2021 16:01...	14.05.2021 16:01...	14.05.2021 16:01...	1835180
Export Docu...	FINISHED	4375...	1-16	ljudmila.petko...	Done, duratio...	0	22.04.2021 11:01...	22.04.2021 11:01...	22.04.2021 11:01...	1800646
Export Docu...	FINISHED	4375...	4	ljudmila.petko...	Done, duratio...	0	22.04.2021 11:00...	22.04.2021 11:00...	22.04.2021 11:00...	1800645
Export Docu...	FINISHED	-1		ljudmila.petko...	Done, duratio...	0	11.04.2021 22:54...	11.04.2021 22:54...	12.04.2021 01:18...	1776006
Export Docu...	FINISHED	5029...	13-192	ljudmila.petko...	Done, duratio...	0	11.04.2021 17:12:...	11.04.2021 17:12:...	11.04.2021 17:16:...	1775550
Export Docu...	FINISHED	5029...	18-166	ljudmila.petko...	Done, duratio...	0	07.04.2021 17:40...	07.04.2021 17:40...	07.04.2021 17:43:...	1766849
Export Docu...	FINISHED	5029...	11-19...	ljudmila.petko...	Done, duratio...	0	02.04.2021 17:4...	02.04.2021 17:4...	02.04.2021 17:47:...	1757125
Export Docu...	FINISHED	361691	1-75	ljudmila.petko...	Done, duratio...	0	31.03.2021 15:46...	31.03.2021 15:46...	31.03.2021 15:47...	1753270
Export Docu...	FINISHED	361691	8	ljudmila.petko...	Done, duratio...	0	31.03.2021 15:41...	31.03.2021 15:41...	31.03.2021 15:41...	1753259

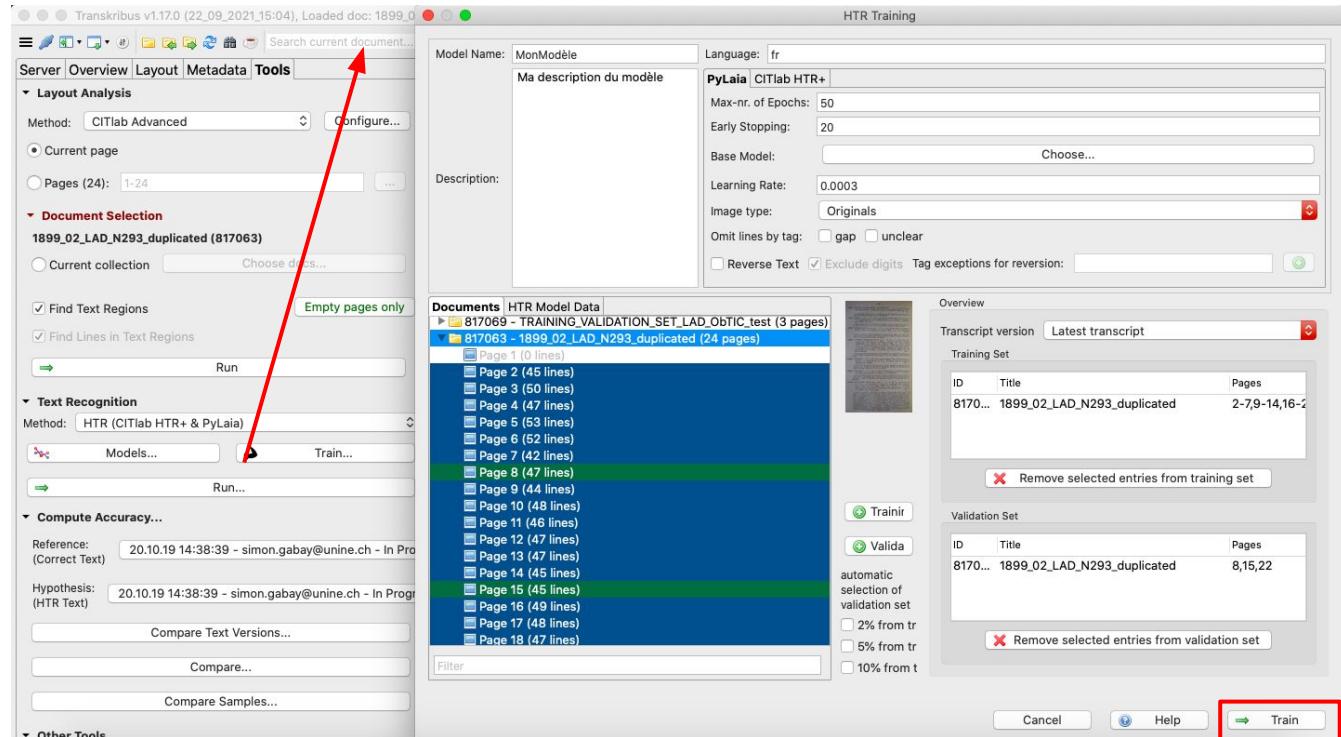
Résultats de transcription

- Transcriptions corrigéables ligne par ligne
- Chaque ligne du texte = ligne segmentée de l'image
- Plusieurs segmentations (*line, baseline, word...*)



Entraînement d'un modèle

- *Train*
- Diviser le jeu de données en *train* et *val*
- Ajuster les paramètres (nº d'itérations, taux d'apprentissage ...)
- Entraîner le modèle



Entraînement d'un modèle

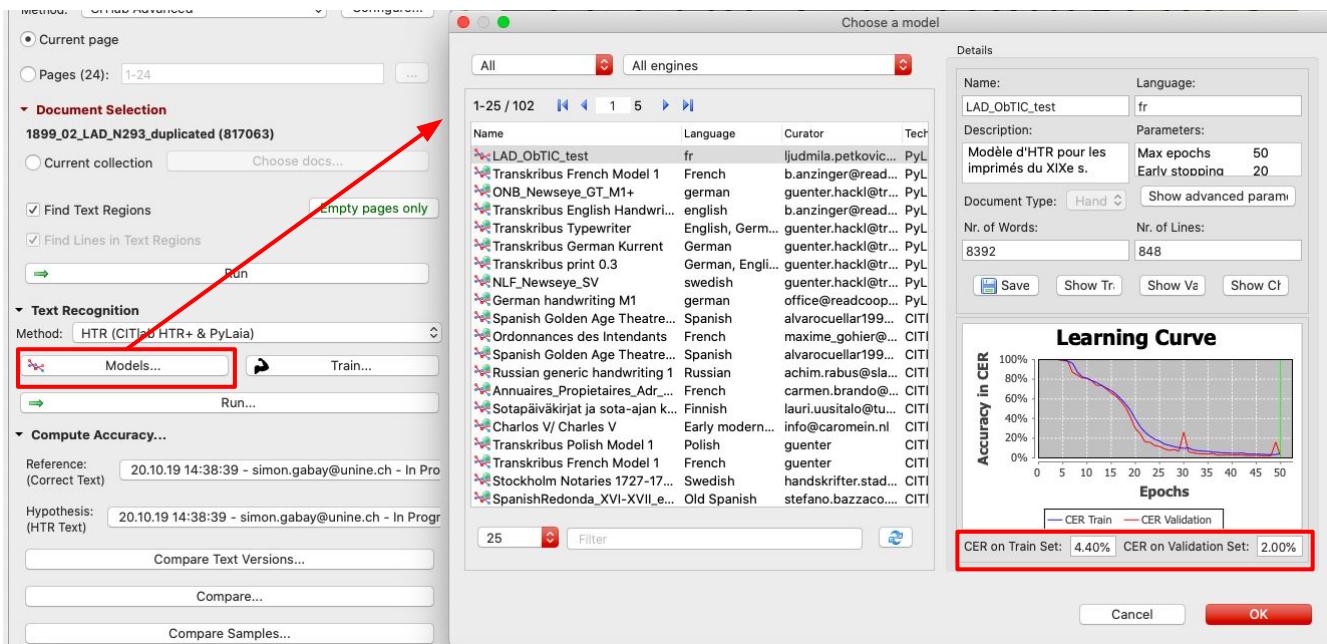
The screenshot shows a software interface with a toolbar at the top containing various icons. A red box highlights the icon for 'Show jobs' (a coffee cup). Below the toolbar is a menu bar with tabs: Server, Overview, Layout, Metadata, Tools, and Show jobs. The 'Tools' tab is selected. To its right is a search bar labeled 'Search current'. The main window title is 'Jobs on Server'. It contains three buttons: 'Cancel job', 'Undo job', and 'Credits'. Below these are filters for 'State' (set to 'ALL'), 'Doc-Id', and 'Type filter'. A table lists 1-50 / 865 jobs. The columns are: Type, State, Doc-Id, Pages, Username, Description, Errors, Created, Started, Finished, and ID. The data in the table includes various job types like PyLaia Decod..., PyLaia Training, Duplicate Do..., Export Docu..., and Export Docu... with different statuses such as FINISHED or -1, and various timestamps and IDs.

- Suivre l'état d'avancement : Show jobs

Type	State	Doc-Id	Pages	Username	Description	Errors	Created	Started	Finished	ID
PyLaia Decod...	FINISHED	8170...		ljudmila.petko...	Done, duratio...	0	28.10.2021 03:10...	28.10.2021 03:10...	28.10.2021 03:11...	2198242
PyLaia Training	FINISHED	-1		ljudmila.petko...	Done, nEpoch...	0	28.10.2021 02:5...	28.10.2021 02:5...	28.10.2021 03:10...	2198239
Duplicate Do...	FINISHED	2306...		ljudmila.petko...	Done, duratio...	0	28.10.2021 02:4...	28.10.2021 02:4...	28.10.2021 02:4...	2198231
Duplicate Do...	FINISHED	2306...		ljudmila.petko...	Done, duratio...	0	28.10.2021 02:3...	28.10.2021 02:3...	28.10.2021 02:3...	2198225
Export Docu...	FINISHED	5060...	245-...	ljudmila.petko...	Done, duratio...	0	14.05.2021 16:01...	14.05.2021 16:01...	14.05.2021 16:01...	1835180
Export Docu...	FINISHED	4375...	1-16	ljudmila.petko...	Done, duratio...	0	22.04.2021 11:01...	22.04.2021 11:01...	22.04.2021 11:01...	1800646
Export Docu...	FINISHED	4375...	4	ljudmila.petko...	Done, duratio...	0	22.04.2021 11:00...	22.04.2021 11:00...	22.04.2021 11:00...	1800645
Export Docu...	FINISHED	-1		ljudmila.petko...	Done, duratio...	0	11.04.2021 22:54...	11.04.2021 22:54...	12.04.2021 01:18...	1776006
Export Docu...	FINISHED	5029...	13-192	ljudmila.petko...	Done, duratio...	0	11.04.2021 17:12:...	11.04.2021 17:12:...	11.04.2021 17:16:...	1775550
Export Docu...	FINISHED	5029...	18-166	ljudmila.petko...	Done, duratio...	0	07.04.2021 17:40...	07.04.2021 17:40...	07.04.2021 17:43:...	1766849
Export Docu...	FINISHED	5029...	11-19...	ljudmila.petko...	Done, duratio...	0	02.04.2021 17:4...	02.04.2021 17:4...	02.04.2021 17:47:...	1757125
Export Docu...	FINISHED	361691	1-75	ljudmila.petko...	Done, duratio...	0	31.03.2021 15:46...	31.03.2021 15:46...	31.03.2021 15:47...	1753270
Export Docu...	FINISHED	361691	8	ljudmila.petko...	Done, duratio...	0	31.03.2021 15:41...	31.03.2021 15:41...	31.03.2021 15:41...	1753259

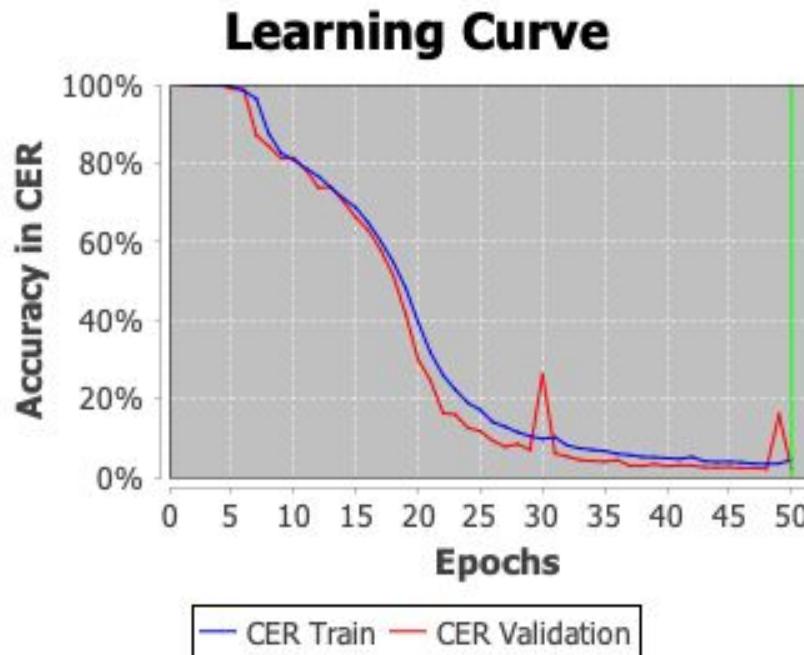
Évaluation d'un modèle

- *Models > Voir les performances du modèle, notamment le CER (Character Error Rate — taux d'erreur de caractère)*



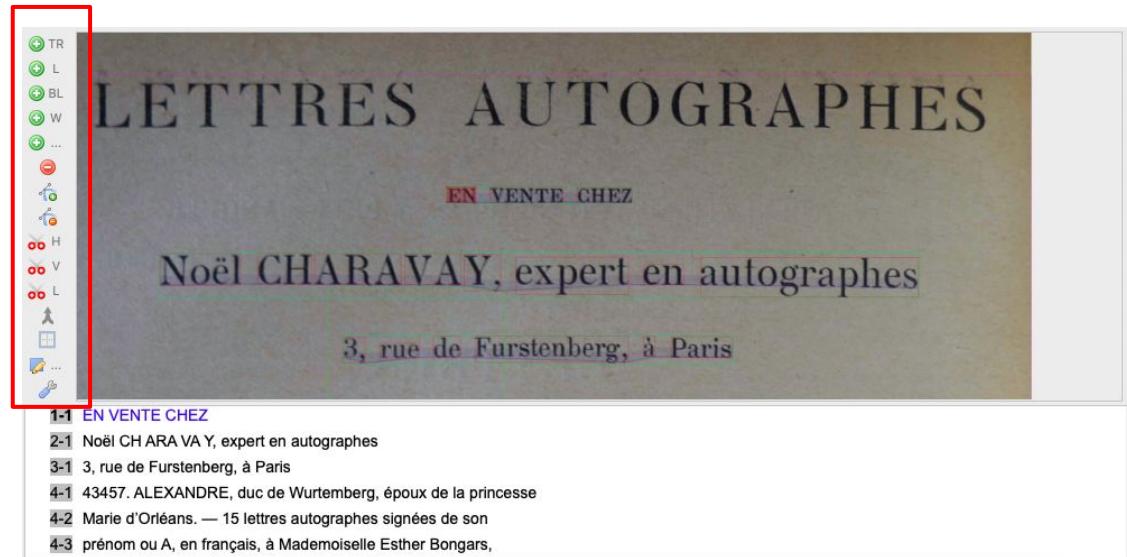
Évaluation d'un modèle

- CER sur les données *train* : 4.40%
- CER sur les données *val* : 2.00%
- L'idéal : <1%
- En l'occurrence, il faudrait ré-entraîner le modèle et le rendre plus performant
- La parution des « pics » vers la fin de l'apprentissage n'est pas souhaitable



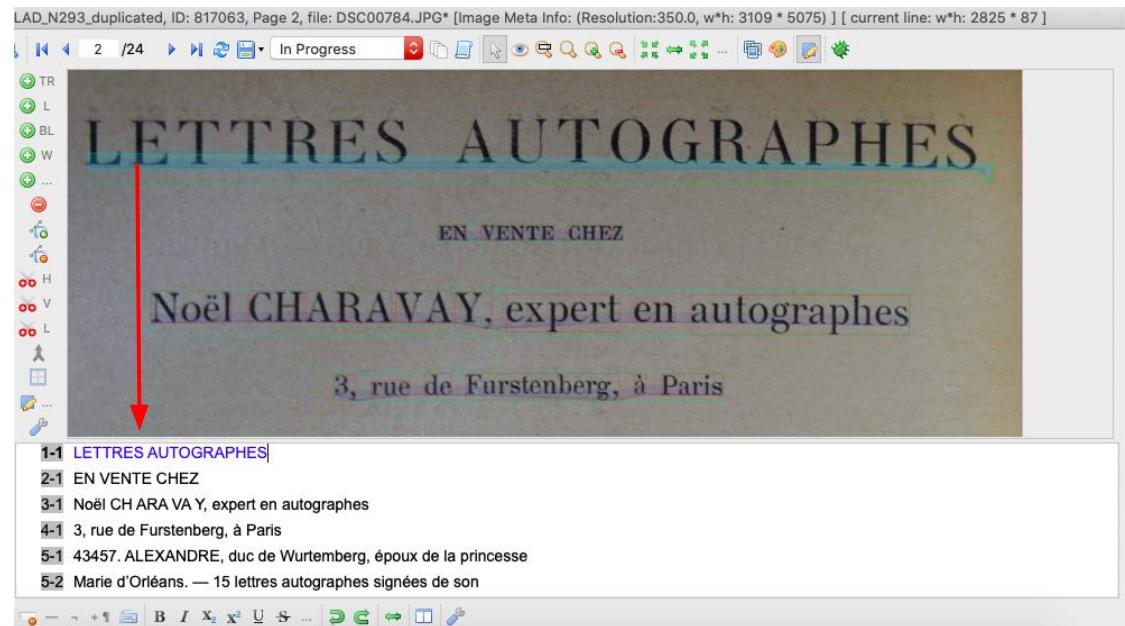
Segmentation

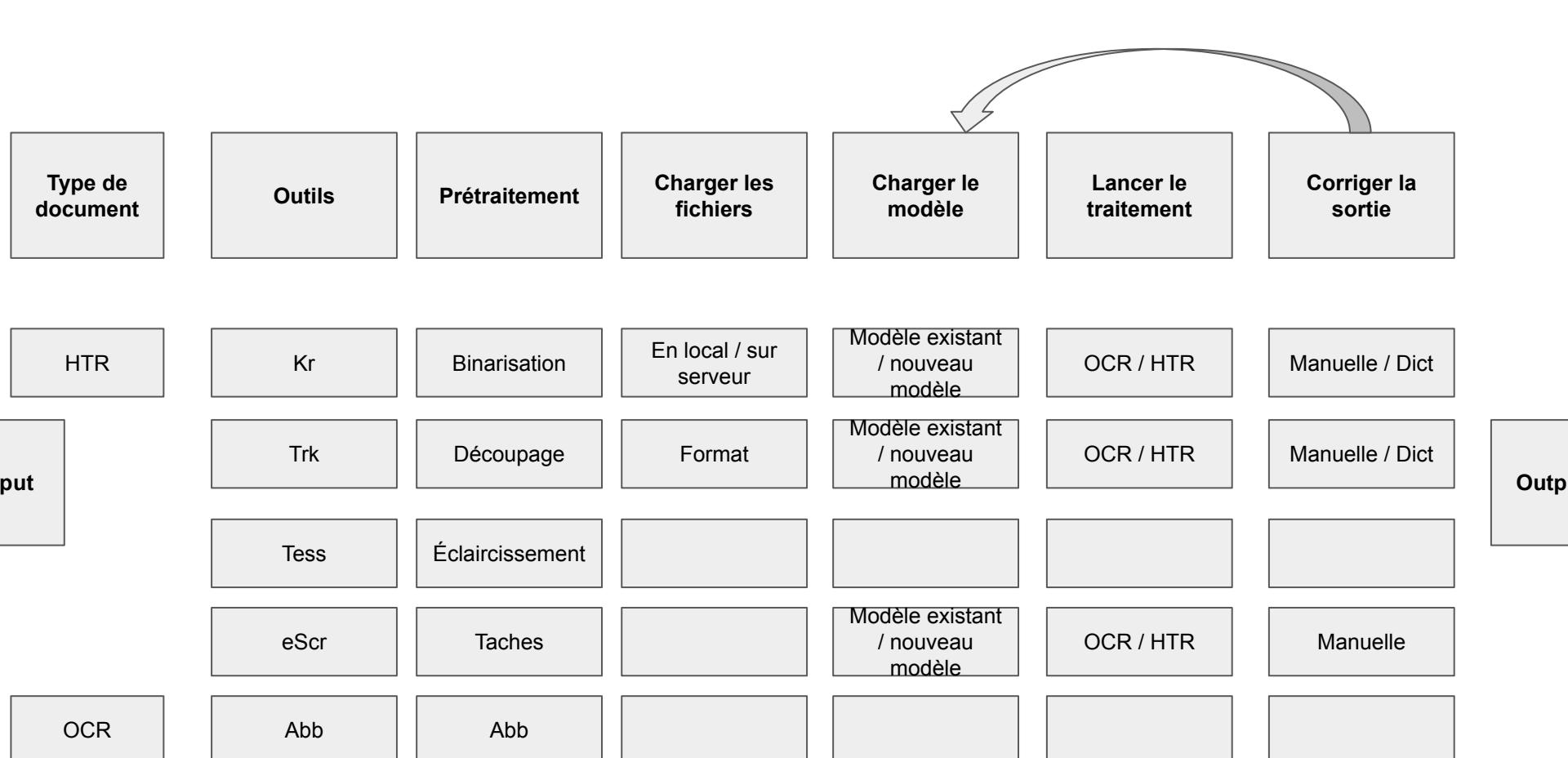
- Admettons que la ligne « LETTRES AUTOGRAPHES » n'est pas segmenté et donc pas transcrise (en l'occurrence, la première ligne du texte « EN VENTE CHEZ » ne correspond pas à la première ligne de l'image)
- Dans ce cas, c'est possible de segmenter manuellement la partie manquante en utilisant les outils à gauche de l'image



Segmentation

- Avec l'outil de segmentation *baseline (BL)*, nous avons segmenté la toute première ligne de l'image
- La segmentation de l'image génère une nouvelle ligne vide dans l'éditeur de texte, où c'est dorénavant possible d'y ajouter les transcriptions manuelles manquantes





Fin