

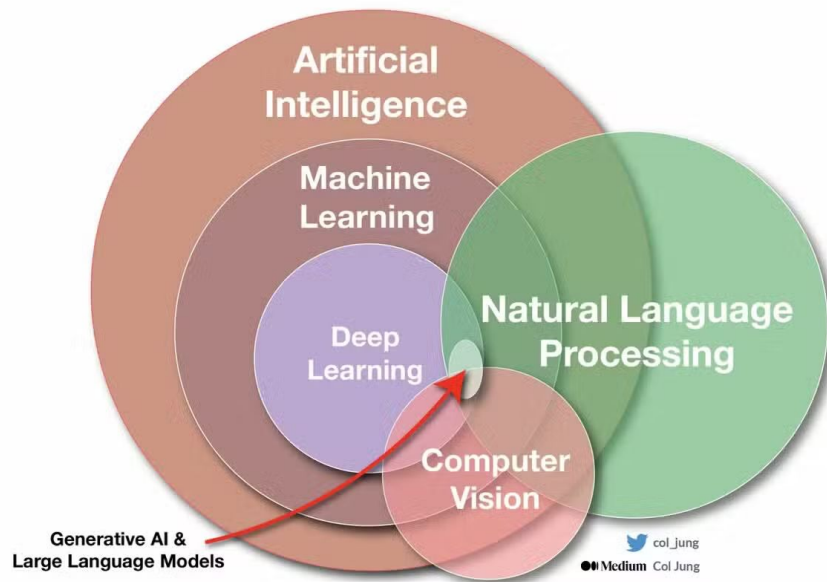
Usages de l'IA à la MSHB

Plateforme Humanités numériques & Plateforme universitaire de données - Bretagne (PUD-B)

Chloé Choquet – Ingénieure en traitement et analyse de corpus textuels – chloe.choquet@mshb.fr

Paul Pinard – Ingénieur-statisticien – paul.pinard@mshb.fr

Intelligence artificielle



<https://www.orsys.fr/orsys-lemag/machine-learning-deep-learning-ia-differences/>

Intelligence artificielle : technique qui permet à un ordinateur de raisonner, planifier, créer et accomplir des tâches qui nécessitent normalement l'intelligence humaine

Machine Learning (apprentissage automatique) : l'ordinateur apprend à partir de données pour faire des prédictions

Deep Learning (apprentissage profond) : machine learning avec des réseaux de neurones pour gérer des données complexes

Computer Vision (vision par ordinateur) : l'ordinateur apprend à comprendre des images ou des vidéos

Traitement automatique des langues (NLP/TAL) : techniques qui permettent à un ordinateur de comprendre, analyser et produire du texte ou de la parole, en combinant informatique, linguistique et intelligence artificielle

LLM (grand modèle de langage) : modèle d'IA capable de comprendre et produire du texte à grande échelle

IA générative : type d'IA qui crée du contenu nouveau, comme du texte, des images ou de la musique



Traitement automatique des langues & Python

Bibliothèques Python : boîtes à outils de fonctions prêtes à l'emploi pour programmer plus vite

Permettent de : tokeniser, lemmatiser, raciniser, attribuer des étiquettes grammaticales, reconnaître des entités nommées, analyser des sentiments, générer des textes, résumer des textes, traduire automatiquement, etc.

- **règles statistiques simples**
rapide (pour petites tâches), capacités basiques
- **modèles probabilistes (machine learning)**
très rapide, bonnes capacités
- **LLM (deep learning)**
lent (pour grandes tâches), très hautes capacités

NLTK

spaCy



Transformers

```

1  # Analyse d'une critique
2
3  import spacy
4  from transformers import pipeline
5
6  # 1 : Analyse linguistique avec spaCy
7  nlp = spacy.load("fr_core_news_md") # modèle français de taille moyenne
8  texte = "Une création sans âme ! Dès les premières minutes, ça sonne faux et j'ai su tout de suite que ça n'allait pas le faire. Les décors alternent, sans la
    moindre harmonie, entre fonds verts criards, CGI dégueulasses – je préfère ne pas insister sur les loups (apparaissant lors de l'épisode avec le vieil aveugle !),
    avec des mouvements saccadés dignes d'un jeu de PlayStation 2 – et quelques extérieurs réels, créant ainsi un patchwork visuel sans cohérence. Le bateau pris dans
    les glaces du prologue – avec ses effets numériques soulignant que Hollywood a fortement régressé dans ce domaine depuis de trop nombreuses années – n'aide pas à
    entrer dans le film . Pas plus que cette étrange idée de faire parler Victor jeune et sa mère en français – une langue qu'aucun des deux acteurs les interprétant
    ne maîtrise un tant soit peu. La musique ne contribue pas non plus à élever le niveau. Elle tente à plusieurs reprises d'imiter Danny Elfman, mais sans en avoir le
    talent, ni l'originalité, ni la fantaisie, ni la puissance mélodique. Résultat : un accompagnement fade, sans ampleur, qui se contente de souligner bien platement
    sans jamais sublimer. Qui a composé cette merde ? Ah oui, ce nullos ultra-méga-surestimé d'Alexandre Desplat – m'étonne pas. Une IA aurait pris sa place, elle
    aurait fait beaucoup mieux." # extrait d'une critique de Frankenstein : https://www.senscritique.com/film/frankenstein/critique/331909408
9
10 doc = nlp(texte)
11
12 print("\nEntités nommées (spaCy) : ")
13 for ent in doc.ents:
14     print(f"{ent.text} → {ent.label_}")
15
16 # 2 : Analyse de sentiments avec transformers
17 analyse_sentiment = pipeline(
18     "sentiment-analysis",
19     model="nlpTown/bert-base-multilingual-uncased-sentiment" # modèle entraîné sur des critiques de produits et films multilingues
20 )
21
22 resultat = analyse_sentiment(texte)
23
24 print("\nAnalyse de sentiments (transformers) : ")
25 print(resultat)

```

Entités nommées (spaCy) :

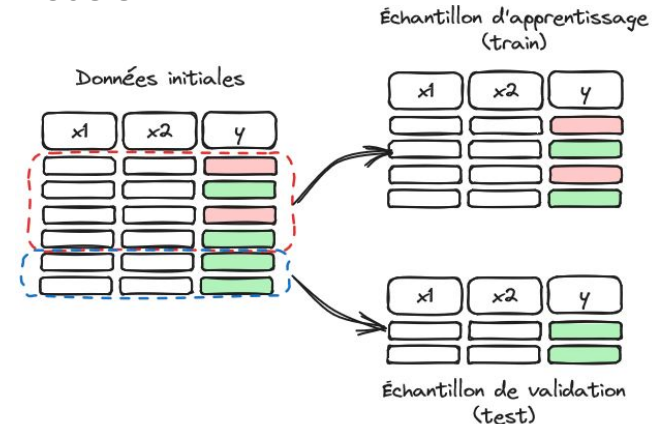
- PlayStation 2 → MISC
- Hollywood → LOC
- Victor → PER
- Danny Elfman → PER
- Ah oui → MISC
- Alexandre Desplat → PER

Analyse de sentiments (transformers) :

```
{'label': '2 stars', 'score': 0.6387473344802856}
```

L'importance des données / Textes :

- Ce que l'on rentre en initialisation impacte grandement le résultat final !
 - Des données qui ne sont pas correctement mises en forme donneront de mauvais résultats
- Méthodes / Codes relativement identique
- Importance surtout sur la **structuration des données**
- Technique courante en apprentissage automatique :
 - $\frac{2}{3}$ de la base de données servira pour faire apprendre un modèle
 - $\frac{1}{3}$ de la base sera pour connaître notre **taux d'erreurs** du modèle



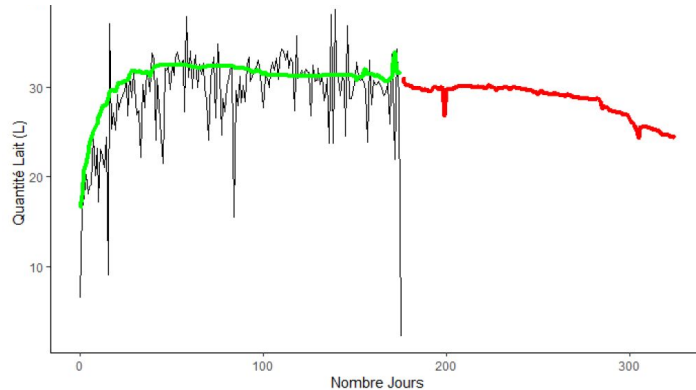
Exemple 1 : Données brutes / Données retouchées

- **Données Brutes**
 - Possède des valeurs aberrantes sur ces variables numériques (Âge, Revenu,...)
 - Possède des “typos” sur ces variables qualitatives (CSP, Situation familiale,...)
- **Essaie de regrouper des individus en fonction :**
 - De leurs tailles, leurs sexes, leurs poids, leurs taille de cheveux, leurs situations familiales...
- **Résultats :**

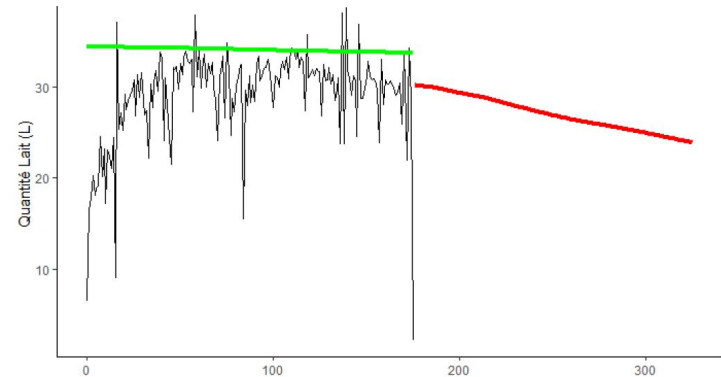
	Données Brutes	Données Retouchées
<i>Précision classification (en %)</i>	~ 90 %	~ 94 %

Exemple 2 : Même données - Résultats différents

Machine Learning :



Deep Learning :



Méthodes	Taux d'erreurs (en moyenne)
Machine Learning 1	7.96 L
Machine Learning 2	5.40 L
Deep Learning	5.69 L

TEMPS	Machine Learning (méthode 1)	Machine Learning (méthode 2)	Deep Learning
Moyenne	2 min 33 sec	10 min 49 sec	1 min 16 sec
Ecart-type	43 sec	16 min 44 sec	59 sec
Maximum	4 min 57 sec	1H 31 min 52 sec	5 min

IR* Huma-Num



hnTools_watchFolder

- > Audio
- ✓> OCR
 - > abbyyCloud
 - > abbyServer
- ✓> Tesseract
 - > toHOCR
 - > toPDF
 - > toTexte
 - > toTSV
- > PDF
- ✓> SpeechToText
 - ✓> whisper
 - > large
 - > medium
 - > small
 - > with_speaker
- > Video

Outils de traitement de Sharedocs :

- **Conversion de la parole en texte** (Speech to Text) : Whisper
- **Reconnaissance de caractères** (OCR) : Abbyy, Tesseract

Ferme de calcul :

L'IR* Huma-num mets à disposition un **serveur de LLM** dédié sous Ollama.

Ollama est un outil qui permet de faire fonctionner des modèles de langage volumineux (LLM) localement sans utiliser de service de cloud.

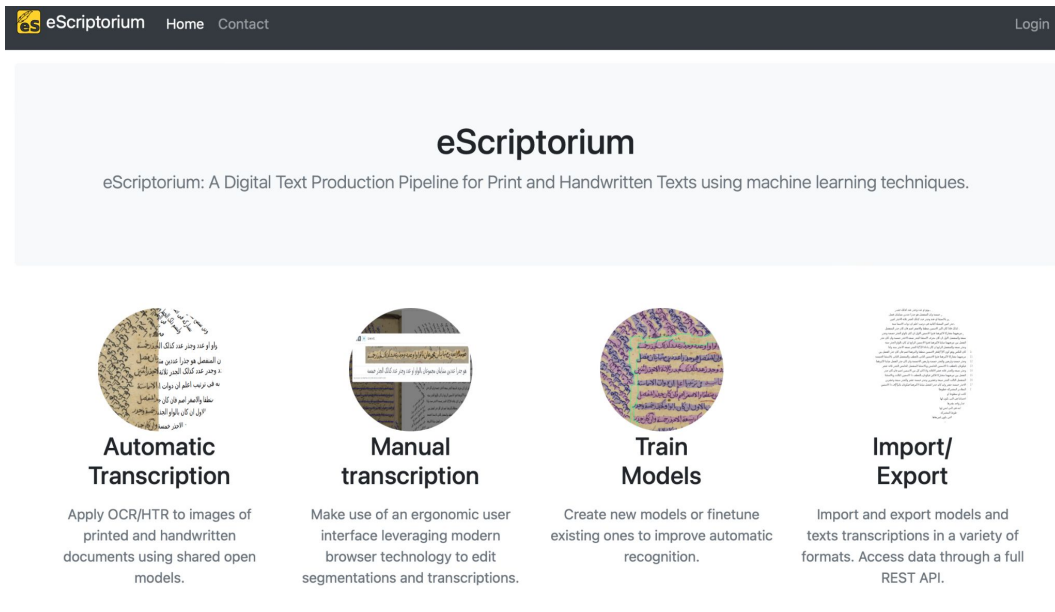
Reconnaissance des textes manuscrits

eScriptorium : plateforme permettant la transcription, l'annotation, la traduction et la publication de documents historiques

→ Possibilité d'entraîner un modèle

 justine.chainiau@mshb.fr

(Chargée d'édition de corpus numériques)



The screenshot shows the eScriptorium website. At the top is a dark navigation bar with the eScriptorium logo, 'Home', 'Contact', and a 'Login' link. The main content area has a light blue background. It features the title 'eScriptorium' and a subtitle: 'eScriptorium: A Digital Text Production Pipeline for Print and Handwritten Texts using machine learning techniques.' Below this are four circular icons representing different features: 'Automatic Transcription' (showing a document with OCR lines), 'Manual transcription' (showing a document with a cursor), 'Train Models' (showing a document with highlighted text), and 'Import/Export' (showing a document with a download icon). Each icon is accompanied by a brief description of the feature.

eScriptorium

eScriptorium: A Digital Text Production Pipeline for Print and Handwritten Texts using machine learning techniques.

- Automatic Transcription**
Apply OCR/HTR to images of printed and handwritten documents using shared open models.
- Manual transcription**
Make use of an ergonomic user interface leveraging modern browser technology to edit segmentations and transcriptions.
- Train Models**
Create new models or finetune existing ones to improve automatic recognition.
- Import/Export**
Import and export models and texts transcriptions in a variety of formats. Access data through a full REST API.

Exemple où l'IA n'est pas utile

NVivo : logiciel d'analyse thématique de contenu

Assistant IA pour l'encodage automatique de thèmes : pas encore au point

Assistant d'encodage automatique - Étape 2 sur 4

Des thèmes ont été identifiés dans vos fichiers. Veuillez examiner les informations fournies ci-après.

Un code ou un cas sera créé pour chaque thème sélectionné.

Thèmes identifiés :

☒	Thèmes	Mentions
☒	professionnelle	621
☒	formation	541
☒	une	518
☒	entreprise	463
☒	accord	354
☒	travail	333
☒	évolution	310
☒	développement	310
☒	mobilité	308
☒	emploi	296
☒	salarié	265
☒	accompagnement	243
☒	durée	218
☒	de travail	216
☒	entretien	213
☒	dispositif	208
☒	projet	208
☒	mois	207
☒	période	207
☒	poste	206
☒	cadre	199
☒	de formation	198

Cliquez sur Terminer pour démarrer l'encodage automatique ou sur Suivant pour plus d'options

Annuler Précédent Suivant Terminer