

INTRODUCTION À L'IA

ACTIVITÉ 2 – DÉCOUVERTE D'UN SYSTÈME RAG COMPLET

PRÉREQUIS

Ces labos ont été réalisés avec l'environnement technique suivant :

- Machine virtuelle ou physique Linux Debian 13 (paquets git et curl) / Python 3.11.2 (paquets python3-pip et python3-venv).
- CPU : dans l'activité 2, les traitements IA sont assez longs (2x20mn en VM avec 2 CPU x 2 coeurs) donc plus c'est mieux ! – Non testé mais les traitements peuvent être optimisés avec un GPU (paramétrage dans les scripts Python).
- RAM : 8 Gio / Stockage minimum : 60 Gio minimum (les modèles sont assez volumineux et téléchargés en local).

PRÉPARATION DE L'ENVIRONNEMENT PYTHON

🔗 Initialiser l'environnement (ou réutiliser celui de l'activité 1) Python3 :

```
mkdir tp-ia-rag
cd tp-ia-rag
python3 -m venv env && source ./env/bin/activate (deactivate pour quitter)
```

RÉCUPÉRATION DU CODE SOURCE

Celui-ci est fourni dans une archive en accompagnement de ce document, sinon il peut être récupéré sur le dépôt Git :

```
git clone https://github.com/yrougy/rpgd-rag
```

TÉLÉCHARGEMENT DU MODÈLE D'EMBEDDING

Étape facultative : le téléchargement de BGE-M3 étant assez volumineux, il est possible de le stocker localement en amont du TP (sinon il est téléchargé au moment où le script l'utilise) :

🔗 Soit depuis le hub Hugging Face (la variable d'environnement permet d'accélérer le téléchargement) :

```
export HF_HUB_DISABLE_XET=1
curl -LsSf https://hf.co/cli/install.sh | bash
hf download BAAI/bge-m3 --local-dir /var/bge-m3
```

🔗 Soit via modelscope :

```
pip install modelscope
modelscope download --model BAAI/bge-m3 --local_dir /var/bge-m3
```

Par la suite, il faudra modifier les scripts Python pour utiliser le modèle local (dans /var/bge-m3).

INSTALLATION DE OLLAMA

🔗 Il faut installer Ollama (outil open-source qui permet d'exécuter des modèles d'intelligence artificielle directement en local sur votre ordinateur, sans passer par le cloud) :

```
curl -fsSL https://ollama.com/install.sh | sh
```

TÉLÉCHARGEMENT DU MODÈLE DE LANGAGE

🔗 Puis télécharger le modèle de langage Llama :

```
ollama pull llama3.1:8b
```

RAG ÉTAPE PAR ÉTAPE

Comme dans l'activité 1, vous pouvez vous aider d'une IA pour répondre aux questions. Vous pouvez également visualiser le code source dans un navigateur (<https://github.com/yrougy/rgpd-rag>) pendant les phases de traitement IA qui peuvent être assez longues selon votre configuration.

Prérequis

Travail à faire 1 Se rendre dans le dossier rgpd-rag et observer le contenu du fichier requirements.txt et relever le nom des 3 bibliothèques nécessaires

 Lancer l'installation (faire la suite pendant ce temps) :

```
cd rgpd-rag
pip install -r requirements.txt
```

Travail à faire 2 Trouver le rôle de chacune des 3 bibliothèques

Étape 1 : chunking

Travail à faire 1 Prendre connaissance du document de référence

 Consulter le contenu du fichier rgpd/rgpd.html

 Donner le nombre de lignes

```
nano rgpd/rgpd.html
wc -l rgpd/rgpd.html
```

Travail à faire 2 Rappeler le rôle de cette étape

 Exécuter cette étape

```
python3 01_chunking.py
```

Travail à faire 3 Observer les lignes 17-18-19 du fichier 01_chunking.py et indiquer leur rôle

Travail à faire 4 Observer les lignes 131-132-133 du fichier 01_chunking.py et indiquer leur rôle

Travail à faire 5 Indiquer le nombre de chunks extraits. Combien de considérants ? Combien d'articles ?

 Observer le contenu du fichier résultat : rgpd_chunks.json

Étape 2 : embedding

Si le modèle BGE-M3 a été téléchargé dans le dossier /var/bge-m3 :

 Modifier le fichier 02_embeddings.py à la dernière ligne :

 Exécuter cette étape (assez longue), donc faire la suite pendant ce temps :

```
python3 02_embeddings.py
```

Travail à faire 6 Rappeler le rôle de cette étape

Travail à faire 7 Donner une définition rapide de BGE-M3 et de son rôle

Étape 3 : visualisation des vecteurs

 Exécuter cette étape :

```
python3 03_view_chromadb.py
```

Travail à faire 8 Relever la dimension des vecteurs et les valeurs minimales et maximales du premier vecteur

Étape 4 : recherche

Si le modèle BGE-M3 a été téléchargé dans le dossier /var/bge-m3 :

 Modifier le fichier 04_recherche.py à la ligne 95 :

Travail à faire 9 Rappeler le rôle de cette étape

Travail à faire 10 Repérer dans le code la question posée

Travail à faire 11 Repérer dans le code la transformation de la question en vecteur

 Exécuter cette étape :

```
python3 04_recherche.py
```

Travail à faire 12 Comparez le score de similarité entre les articles trouvés et l'article aléatoire

Étape 5 : modèle de langage et réponse

Modèle de langage

- ☑ Si ce n'est déjà fait, installer Ollama et le modèle llama3.1:8b (voir « préparation » au début de l'activité)

- ☑ Démarrer le service Ollama :

```
service ollama start
```

- ☑ Vérifier que le modèle llama3.1:8b est présent :

```
ollama ls
NAME          ID          SIZE      MODIFIED
llama3.1:8b    46e0c10c039e 4.9 GB    34 hours ago
```

- ☑ Démarrer le modèle :

```
ollama run llama3.1:8b
```

- ☑ Une invite interactive apparaît, taper « bonjour » pour vérifier que notre « ami » est prêt à travailler

- ☑ Taper /bye pour sortir

```
>>> bonjour
Bonjour ! It's nice to speak in French. Comment allez-vous aujourd'hui ? (How are you today?)[GIN] 2026/02/26 - 07:13:57 | 200 | 12.995829372s | 127.0.0.1 | POST      "/api/chat"

>>> /bye
```

Préparation du code

- ☑ Modifier le fichier 05_ollama_integration.py :

```
ligne 8 : OLLAMA_MODEL = 'llama3.1:8b'
ligne 10 : EMBEDDING_MODEL = '/var/bge-m3'
```

- ☑ Exécuter cette étape (assez longue donc faire la suite pendant ce temps), taper une question (par exemple : quand a été publié le RGPD ?) :

```
python3 05_ollama_integration.py
```

Travail à faire 13 Rappeler le rôle de cette étape

Travail à faire 14 L'étape 2 du script indique : « Construction du prompt avec contexte... ». Donner une définition de « contexte » pour un modèle de langage.

Travail à faire 15 Repérer dans le code du fichier 05_ollama_integration.py quel contexte est fourni au modèle.

Travail à faire 16 :

- a. Donner une définition rapide de llama3.1:8b
- b. Préciser ce que signifie le 8b ?
- c. Rechercher quelles sont les autres versions de llama3.1. Indiquer les conditions matérielles recommandées pour l'exécution (gpu etc.)

Travail à faire 17 Donner la signification de « paramètres » pour un modèle d'IA et donner des éléments de comparaison de ce modèle avec d'autres.