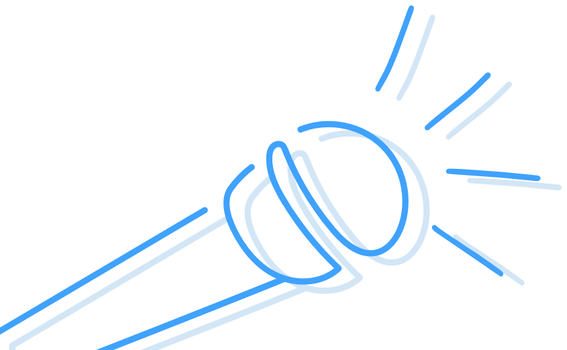
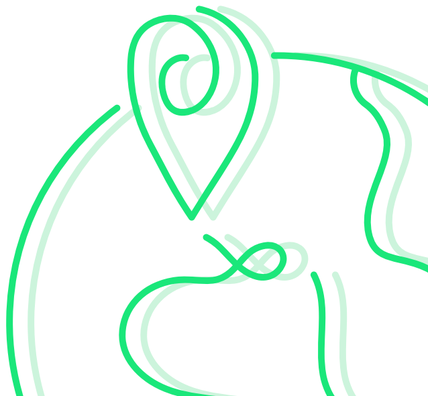

DEVOXX France

# TRANSCENDEZ LES FRONTIÈRES LINGUISTIQUES

## avec des APIs de Machine Learning sur mesure



OVHcloud



# PRÉSENTATION

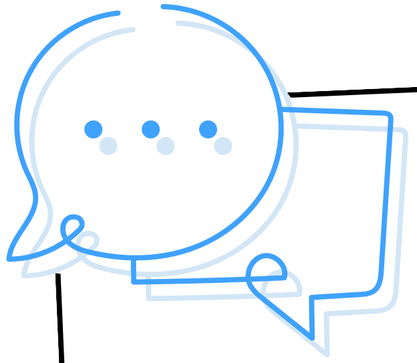
- Machine Learning Engineer
- OVHcloud
- AI Solutions Team

ÉLÉA PETTON

@EleaPetton

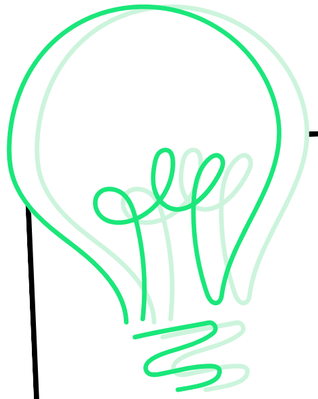
eleapttn

Eléa PETTON



# ÇA VOUS DIT ?

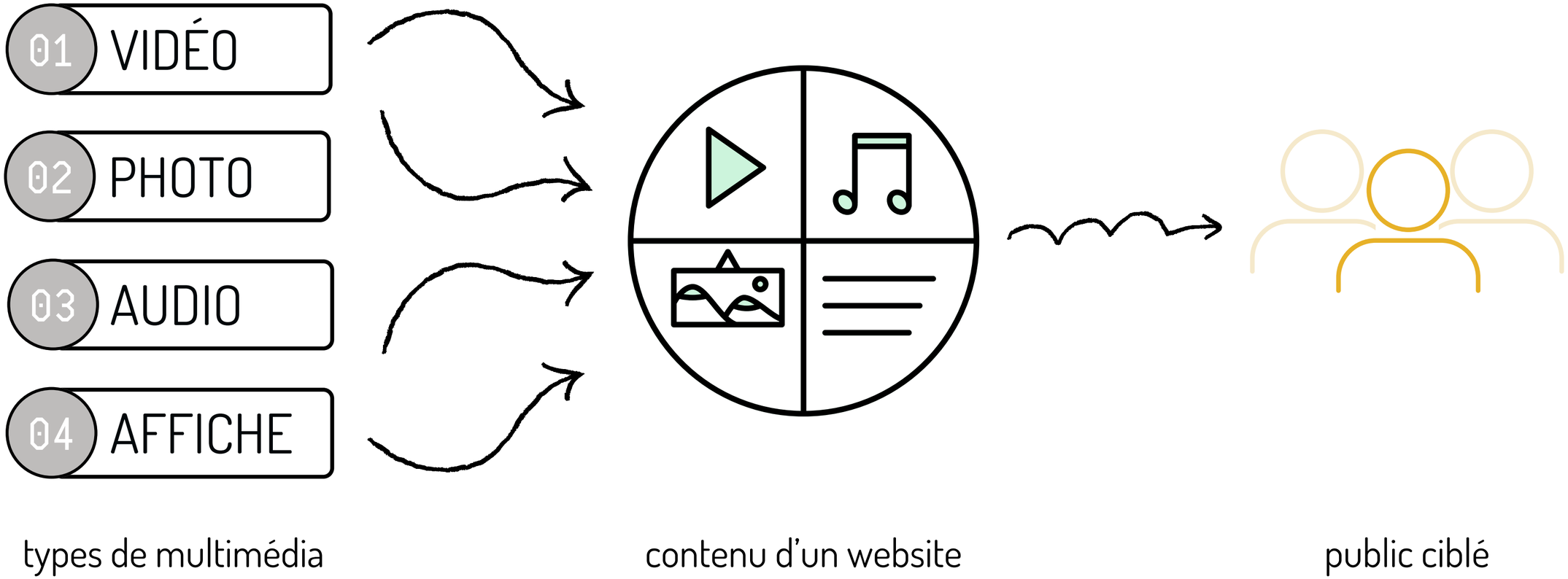
“ Embarquez dans le développement d’une solution de transcription temps réel de vos contenus multimédia... ”



# INTRODUCTION

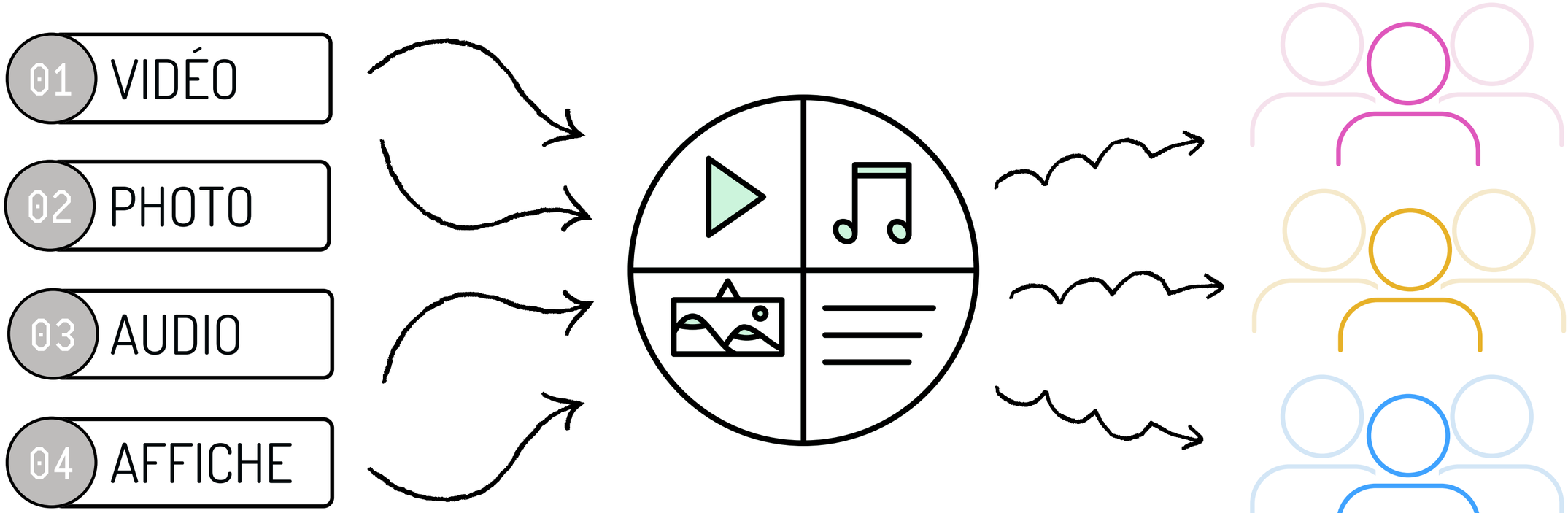
# INTRODUCTION

“ Quels  sont les différents types de contenus multimédia et leur(s) public(s) ? ”



# INTRODUCTION

“ Quels  sont les différents types de contenus multimédia et leur(s) public(s) ? ”




types de multimédia

contenu d’un website

publicS ciblés

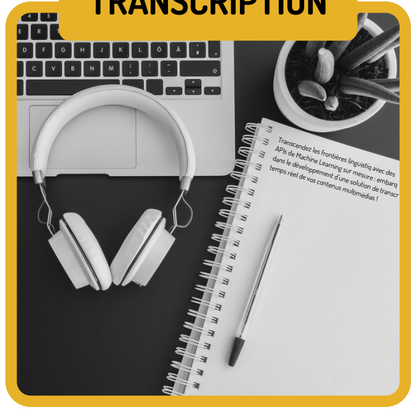
# TRANSFORMER SES CONTENUS MULTIMEDIA, QU'EST-CE QUE ÇA VEUT DIRE ?

## TRADUCTION


**Changer la langue de...**

- sa page web
- son post Twitter, LinkedIn, ...
- ses slides

## TRANSCRIPTION




**Passer de l'oral à l'écrit pour...**

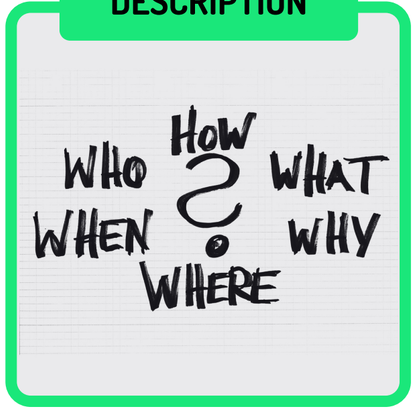
- sous-titrer des vidéos, podcasts
- garder le contenu d'une réunion

## SYNTHÈSE VOCALE

**Passer de l'écrit à l'oral pour...**

- favoriser l'accessibilité
- doubler les voix

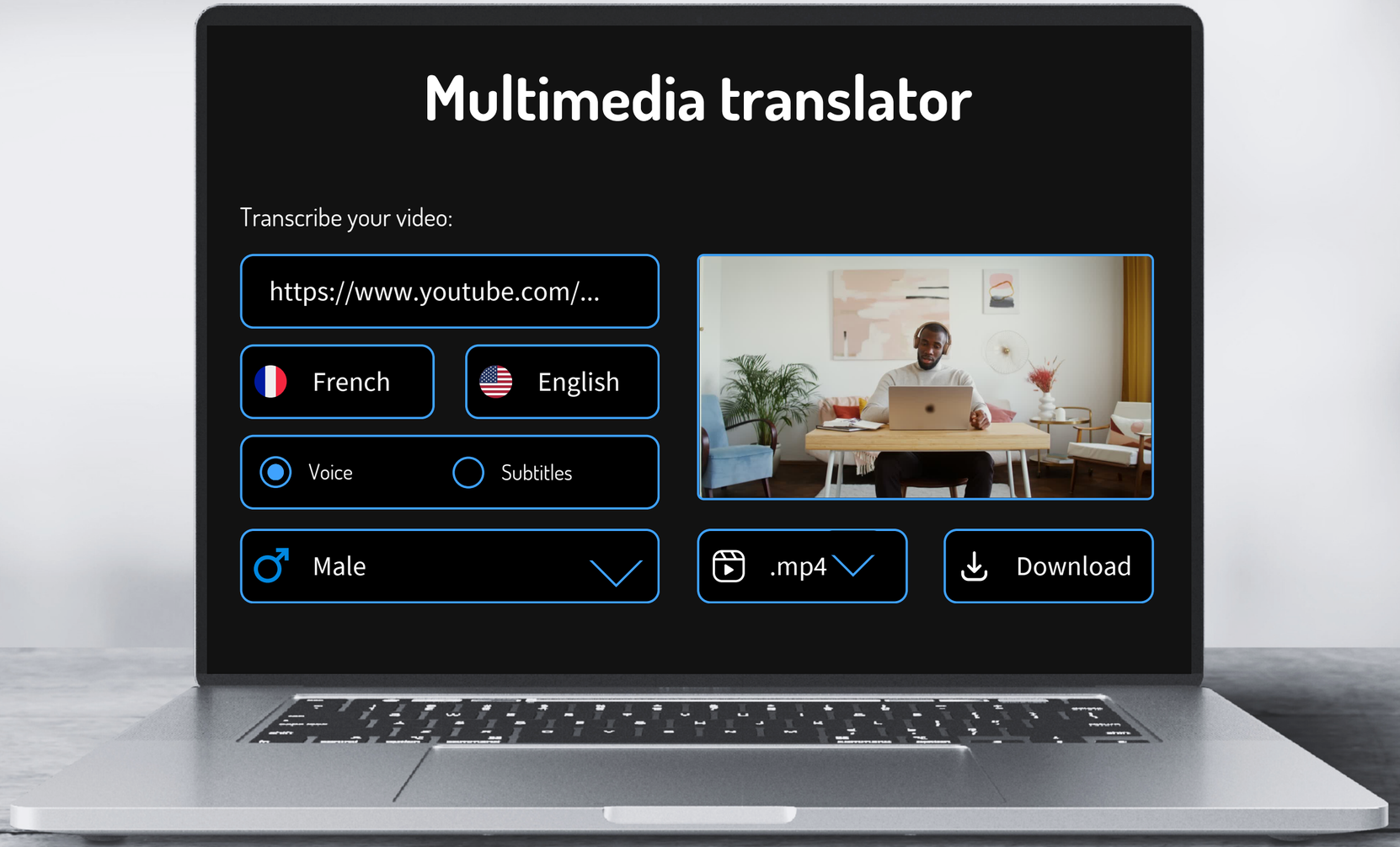
## DESCRIPTION



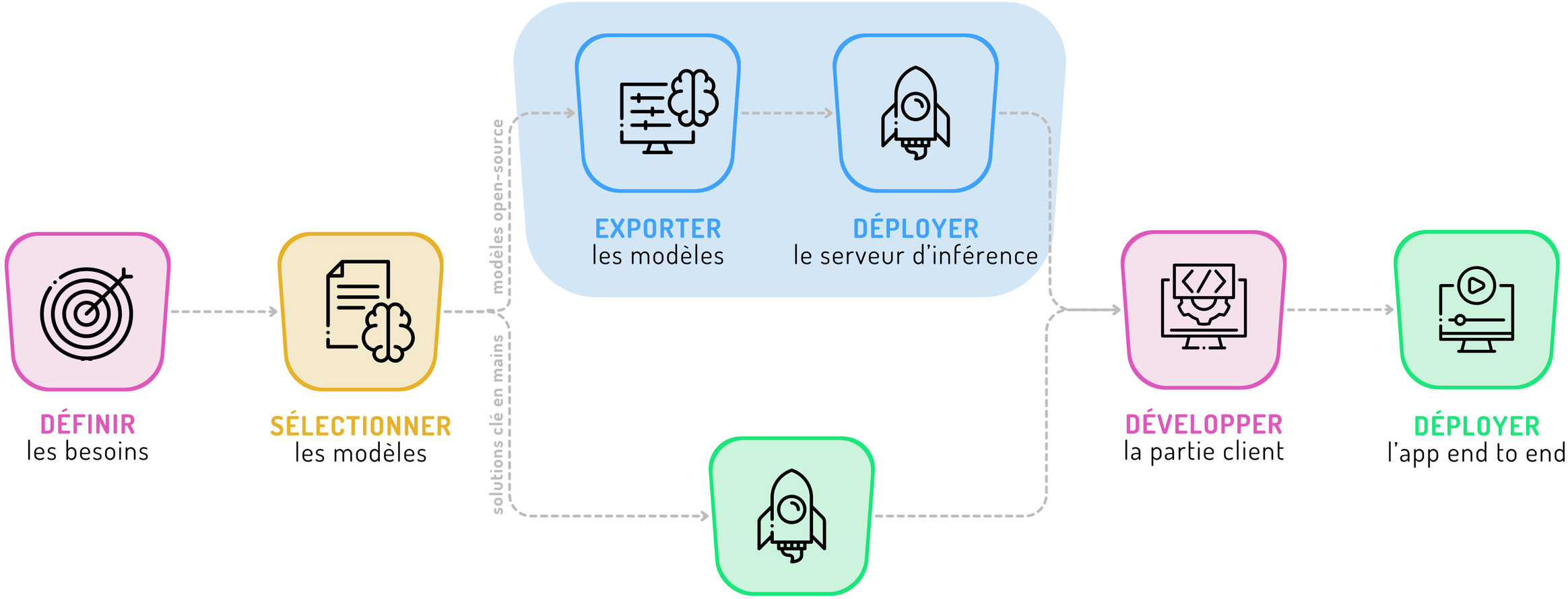

**Décrire ou résumer...**

- une vidéo, un podcast
- le contenu d'une réunion
- une documentation

# ET EN PRATIQUE, ÇA DONNE QUOI ?



# OBJECTIFS


modèles open-source
solutions clé en mains
EXPORTER
les modèles
DÉPLOYER
le serveur d'inférence
DÉFINIR
les besoins
SÉLECTIONNER
les modèles
CHOISIR
ses AI Endpoints
DÉVELOPPER
la partie client
DÉPLOYER
l'app end to end

# DÉFINIR
## les besoins

# QUELS SONT NOS PERSONAE ?

### Transcrire les meetings

- **Retranscrire** les meetings à l'écrit
- **Transcrire** les meetings dans une autre langues
- **Résumer** les meetings

### Se développer à l'international

- **Traduire du contenu multimedia** pour un usage multilingue
- **Adapter les vidéos** pour un public mondial

### Être plus inclusif !

- **Sous-titrer les vidéos** pour les personnes malentendantes
- **Inclure les personnes** qui ne parlent pas la même langue

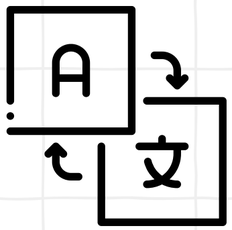
### Élargir son public...

- En tant que créateur de contenu, j'aimerais **élargir ma communauté**
- Créer et publier un **contenu plus attrayant**

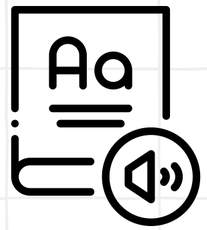
# 3 TÂCHES CIBLES

**AUDIO**
recognition

**TEXT**
translation

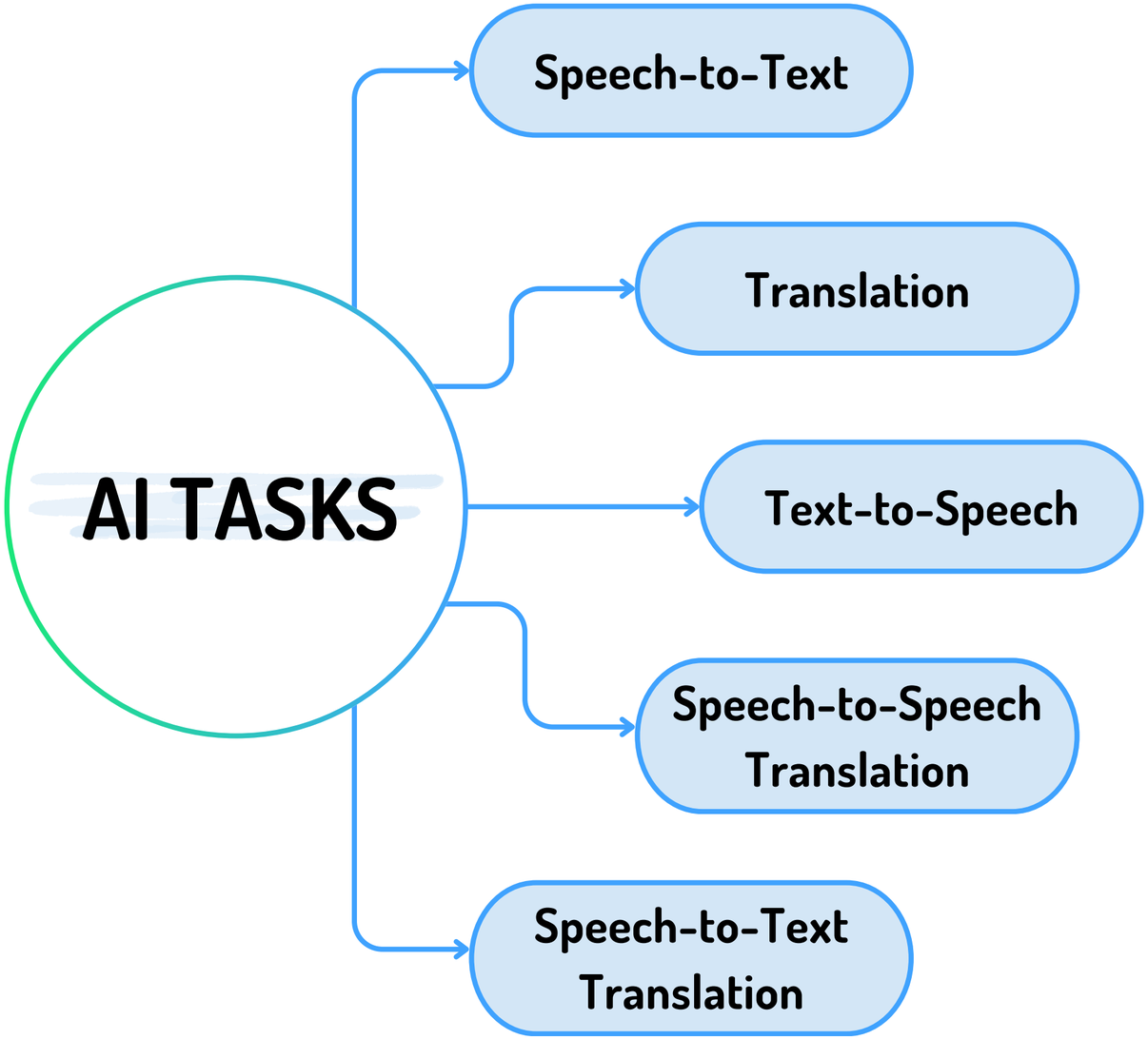



**SPEECH**
synthesis

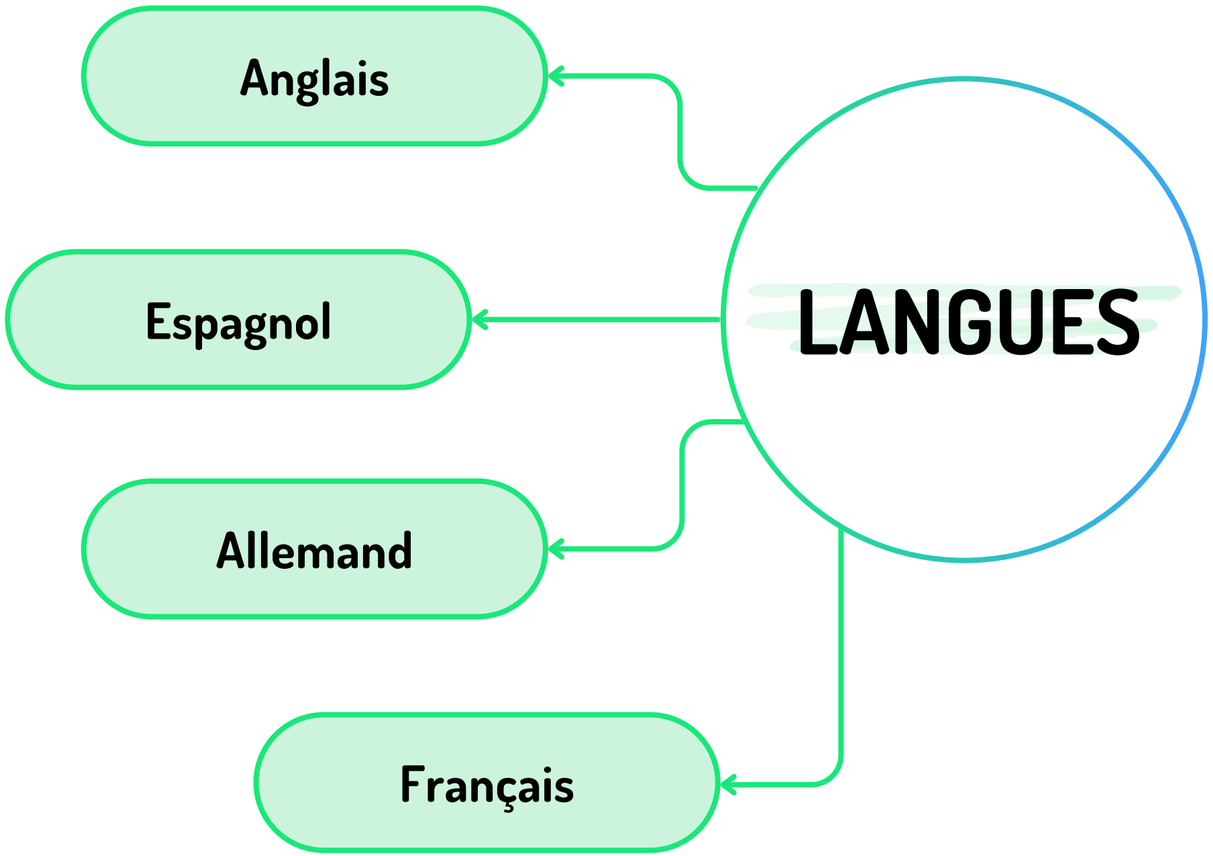


# SÉLECTIONNER
## les modèles

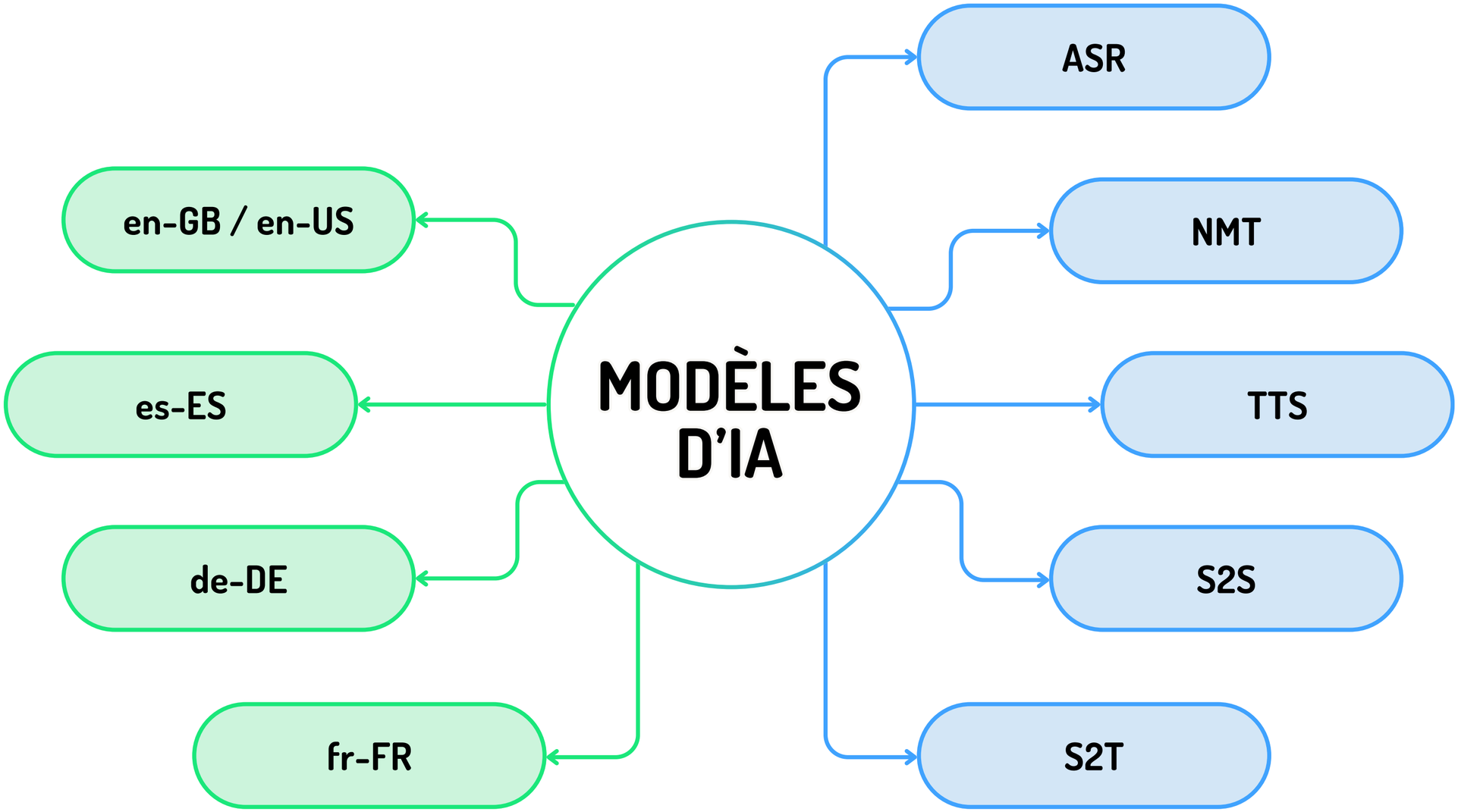
# SÉLECTIONNER LES MODÈLES



# SÉLECTIONNER LES MODÈLES



# SÉLECTIONNER LES MODÈLES



# COMMENT ACCÉDER À CES MODÈLES ?

**Définir le but de ces modèles**

" Le besoin ici n'est pas d'entraîner les modèles mais bien d'en faire l'**inférence**. Cela implique déployer les modèles de manière optimisée : coûts minimum et meilleur ratio précision / latence "

**Prendre des modèles Open Source**

- ⊕ Accès à de nombreux modèles
- ⊕ Possibilité de "personnalisation"
- ⊕ Prédictibilité des coûts

- ⊖ Complexité du déploiement
- ⊖ Gestion du RUN
- ⊖ Licences à non usage commercial

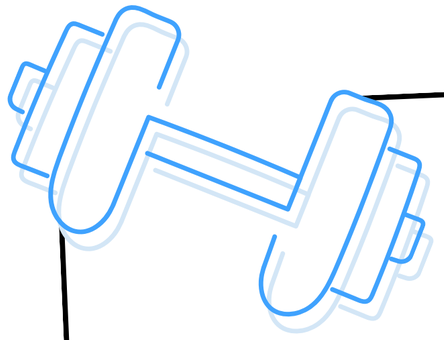
- ✓ Connaissances en ML
- ✓ Notion d'optimisation de modèles
- ✓ Des GPUs à disposition
- .....
- ✓ & du temps !

**Sélectionner des solutions clés en main**

- ⊕ Endpoints API à disposition
- ⊕ Simplicité d'utilisation (call APIs)
- ⊕ Performances optimales
- ⊕ Pas de gestion du RUN

- ⊖ Coût variable - solution "Pay per call"
- ⊖ Peu de possibilité de personnalisation

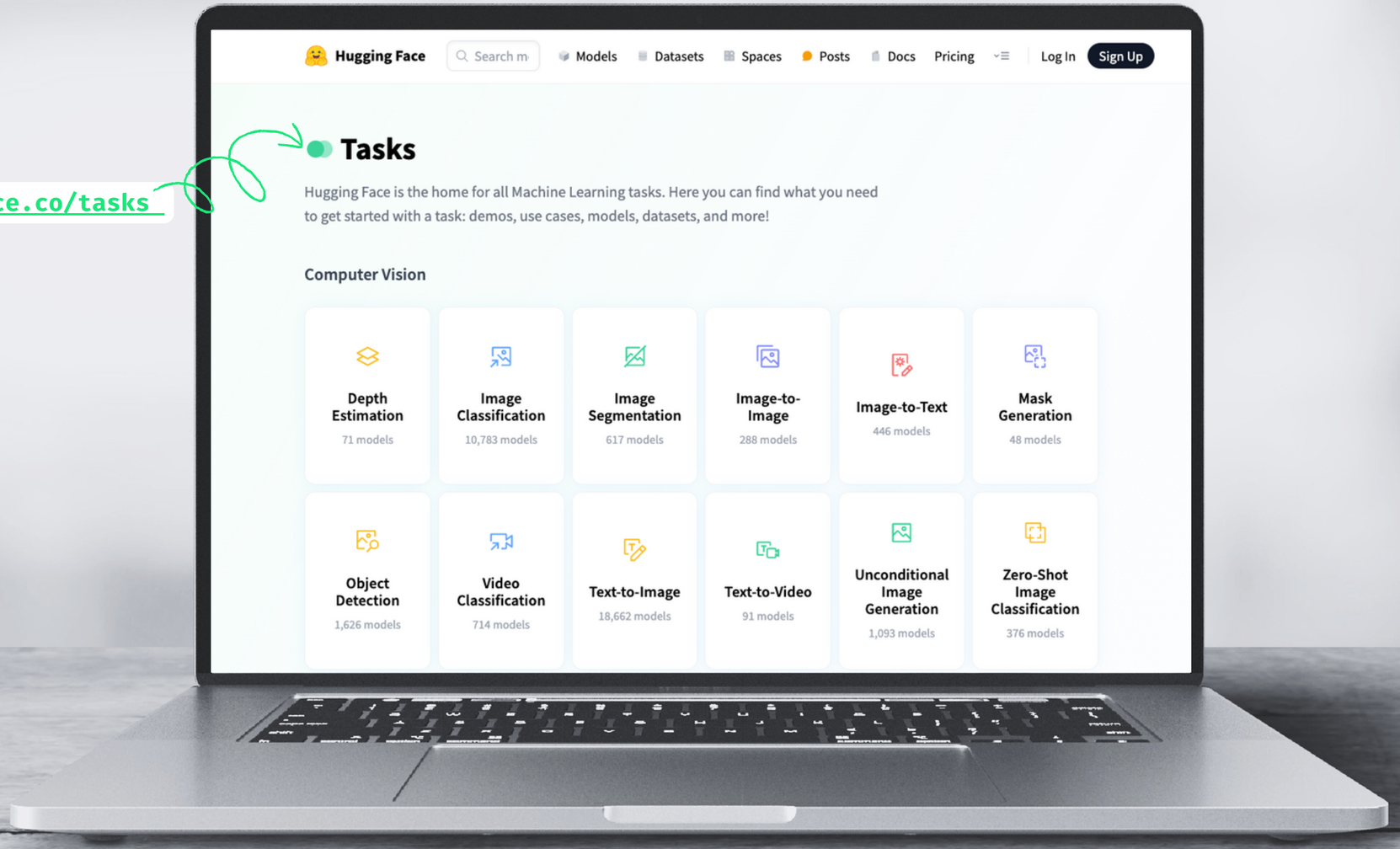
- ✓ Compréhension des paramètres d'entrée du modèle
- ✓ Appropriation des sorties du modèle
- .....
- ✓ & des connaissances en Puzzles !



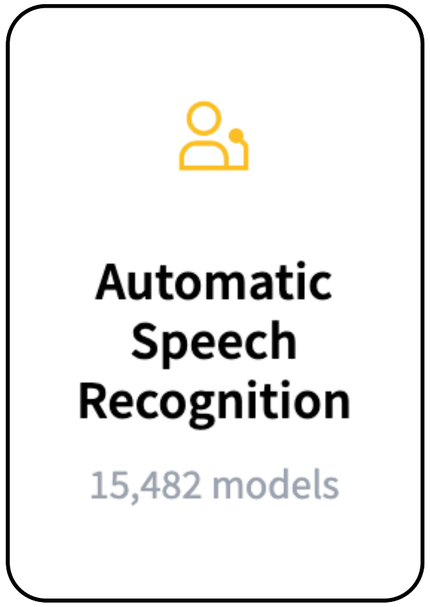
# UTILISER
## l'Open Source

# COMMENT TROUVER CES MODÈLES ?

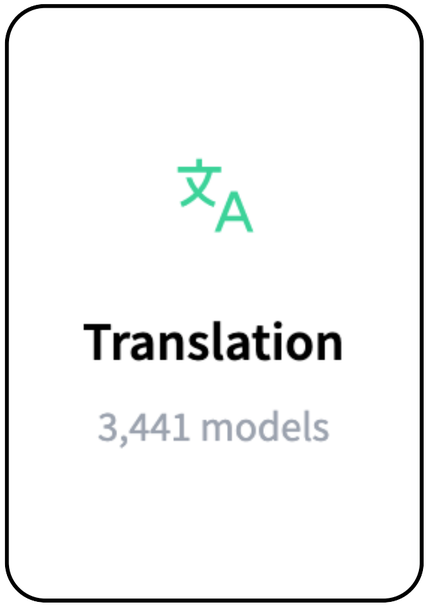
https://huggingface.co/tasks



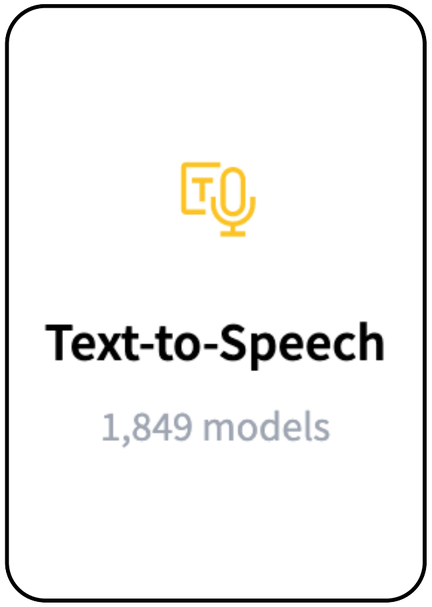
# HUGGING FACE TASKS




ASR

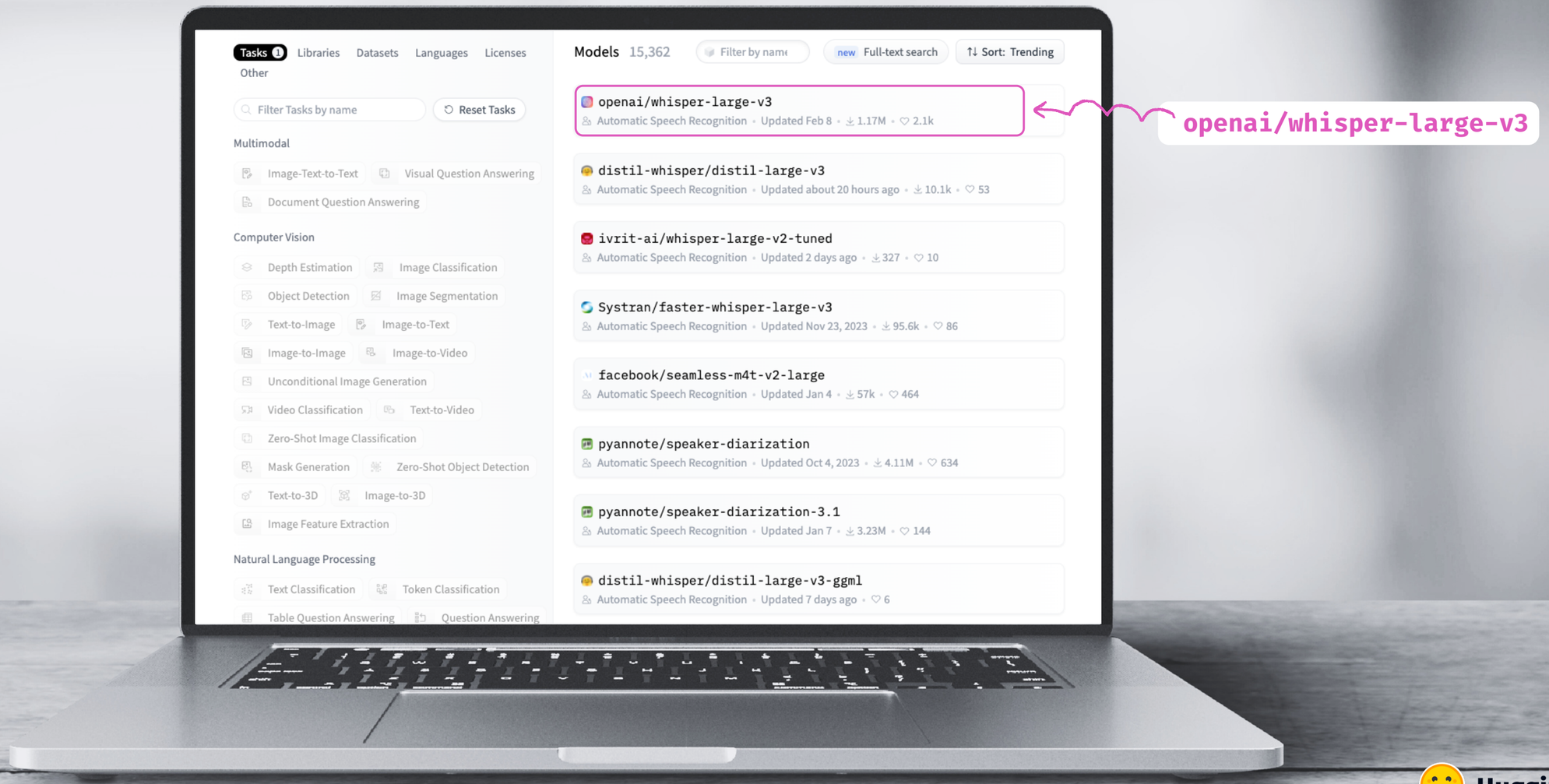



NMT

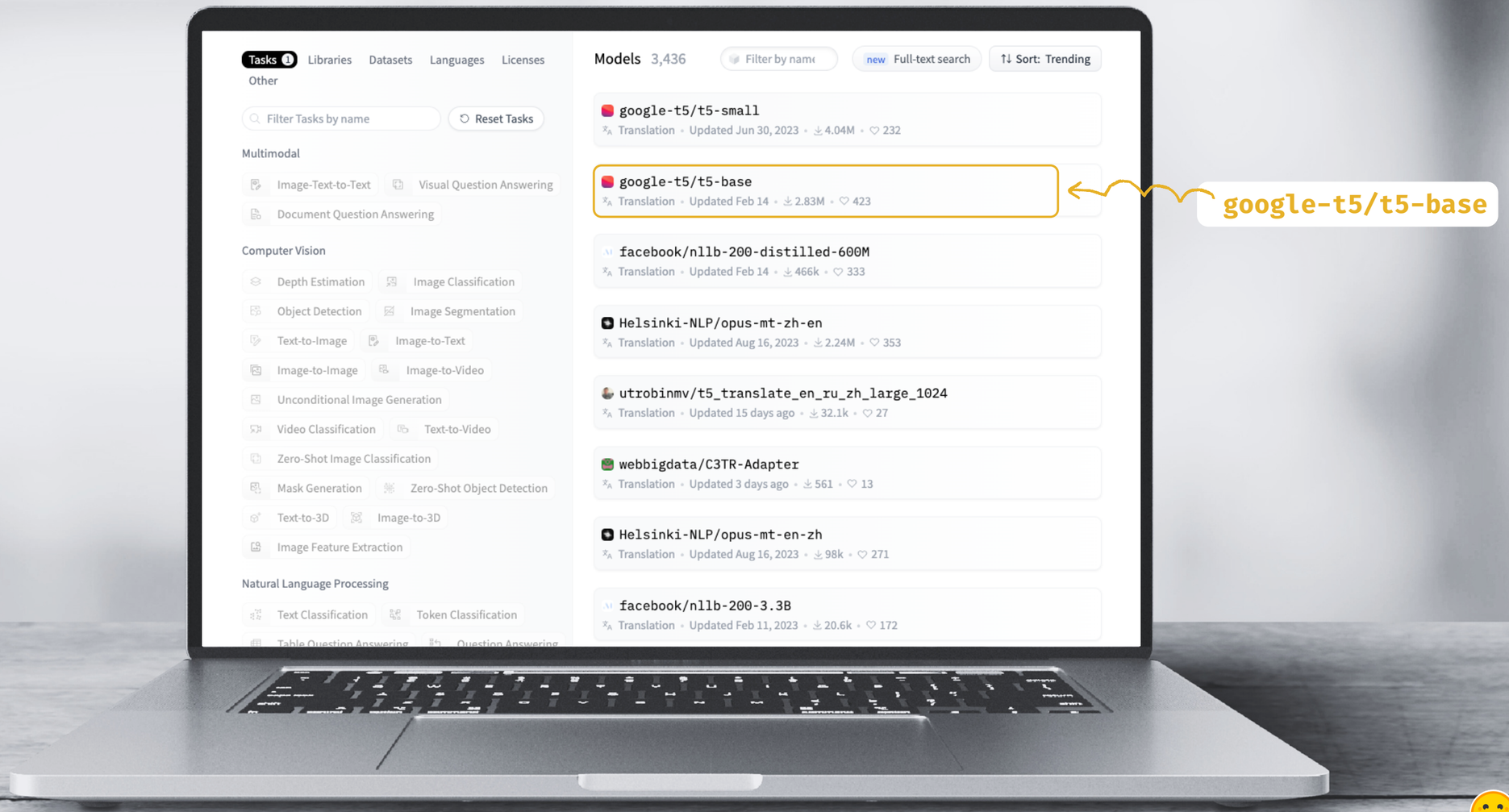



TTS

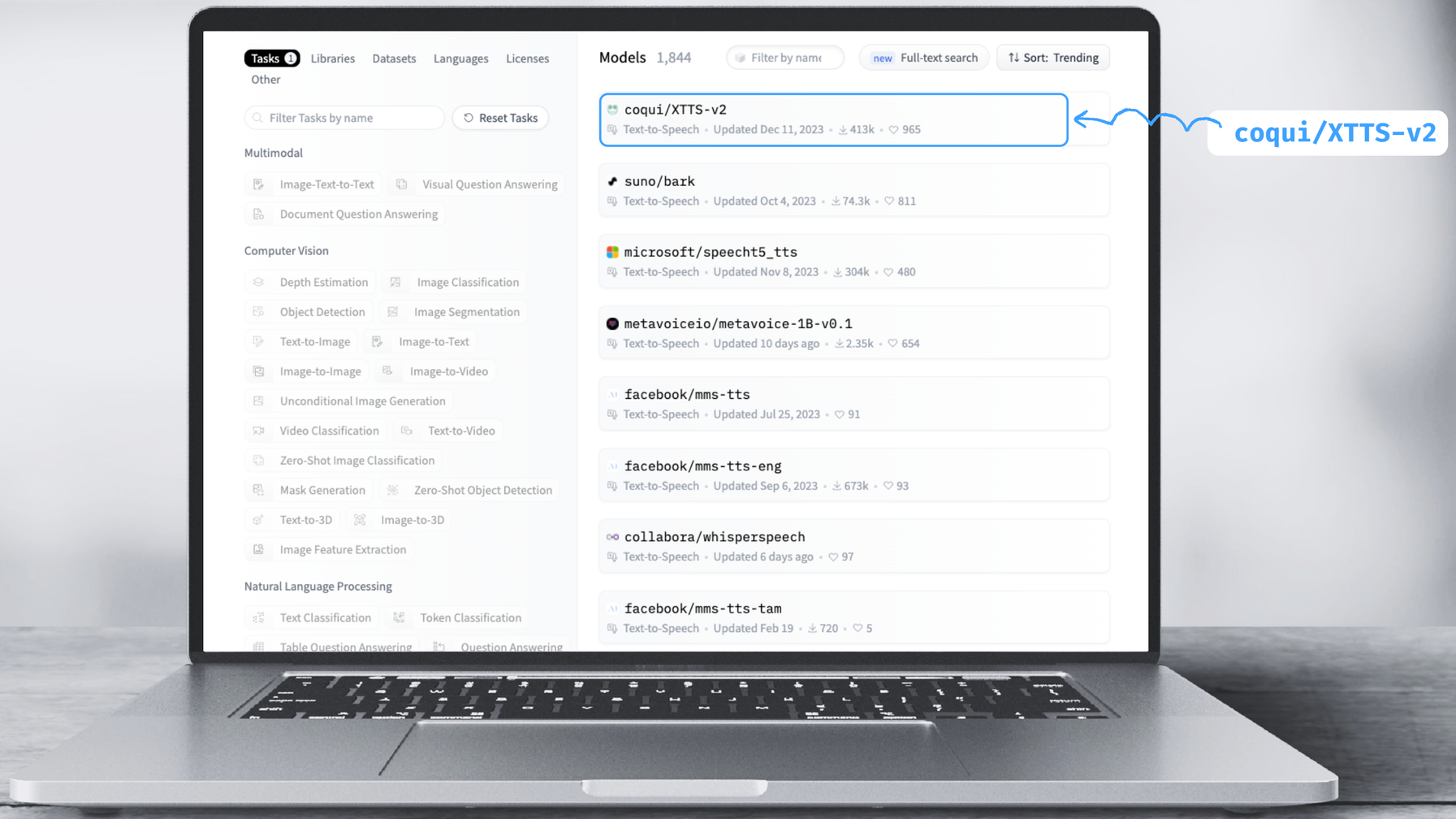
# AUTOMATIC SPEECH RECOGNITION



# TRANSLATION



# TEXT TO SPEECH




coqui/XTTS-v2

# COMMENT DÉPLOYER CES MODÈLES ?

**ACCÉDER**
à un notebook avec GPUs

**TESTER**
le(s) modèle(s) localement

**EXPORTER**
les modèles au format ONNX

**DÉVELOPPER**
les fonctions de pré/post processing

**FORMATER**
le code et le repo de modèles

**STOCKER**
les modèles dans un bucket S3

**DÉPLOYER**
le serveur d'inférence dans le cloud

# COMMENT DÉPLOYER CES MODÈLES ?

**ACCÉDER**
à un notebook avec GPUs

**TESTER**
le(s) modèle(s) localement

**EXPORTER**
les modèles au format ONNX

**DÉVELOPPER**
les fonctions de pré/post processing

**FORMATER**
le code et le repo de modèles

**STOCKER**
les modèles dans un bucket S3

**DÉPLOYER**
le serveur d'inférence dans le cloud

# ACCÉDER
## à un notebook

# ACCÉDER À UN NOTEBOOK AVEC GPUs


**RESSOURCES DE CALCUL**
- 1 GPU
- V100S / A100 / H100

**JUPYTER NOTEBOOK**
- JupyterLab editor
- Prepackaged environment

**ACCÈS SÉCURISÉ**
- Private mode
- Personal token access

# OVHcloud - AI NOTEBOOKS

| | | |
|---|---|---|
| 01 | NOTEBOOK AS A SERVICE | Nptebooks JupyterLab ou VS Code préconfigurés |
| 02 | RESSOURCES DE CALCUL | Le notebook est lancé dans le cloud avec GPU (ou CPU) |
| 03 | MODE DE FACTURATION | Le client est facturé à la minute utilisée |
| 04 | ENVIRONNEMENT PRÉ-PACKAGÉ | Frameworks ML disponibles : PyTorch, TensorFlow, Transformers, ... |
| 05 | PERSISTANCE DU WORKSPACE | Conservation du workspace de l'utilisateur entre les exécutions pour conserver les packages installés |

# COMMENT DÉPLOYER CES MODÈLES ?

**ACCÉDER**
à un notebook avec GPUs

**TESTER**
le(s) modèle(s) localement

**EXPORTER**
les modèles au format ONNX

**DÉVELOPPER**
les fonctions de pré/post processing

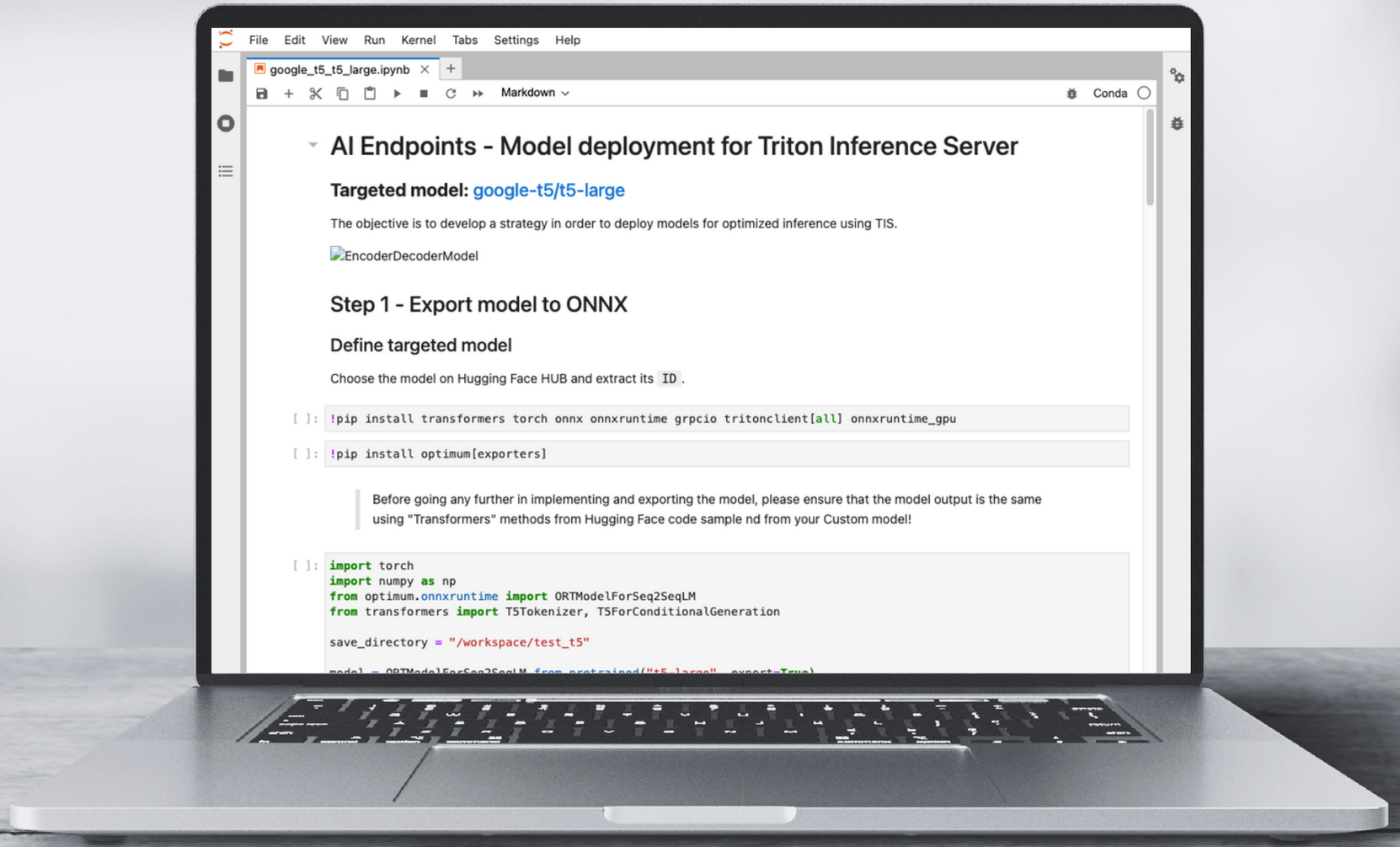
**FORMATER**
le code et le repo de modèles

**STOCKER**
les modèles dans un bucket S3

**DÉPLOYER**
le serveur d'inférence dans le cloud

# TESTER
## les modèles

# TESTER LES MODÈLES LOCALEMENT



# COMMENT DÉPLOYER CES MODÈLES ?

**ACCÉDER**
à un notebook avec GPUs

**TESTER**
le(s) modèle(s) localement

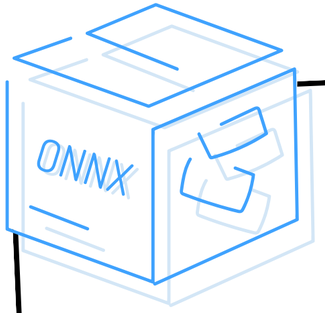
**EXPORTER**
les modèles au format ONNX

**DÉVELOPPER**
les fonctions de pré/post processing

**FORMATER**
le code et le repo de modèles

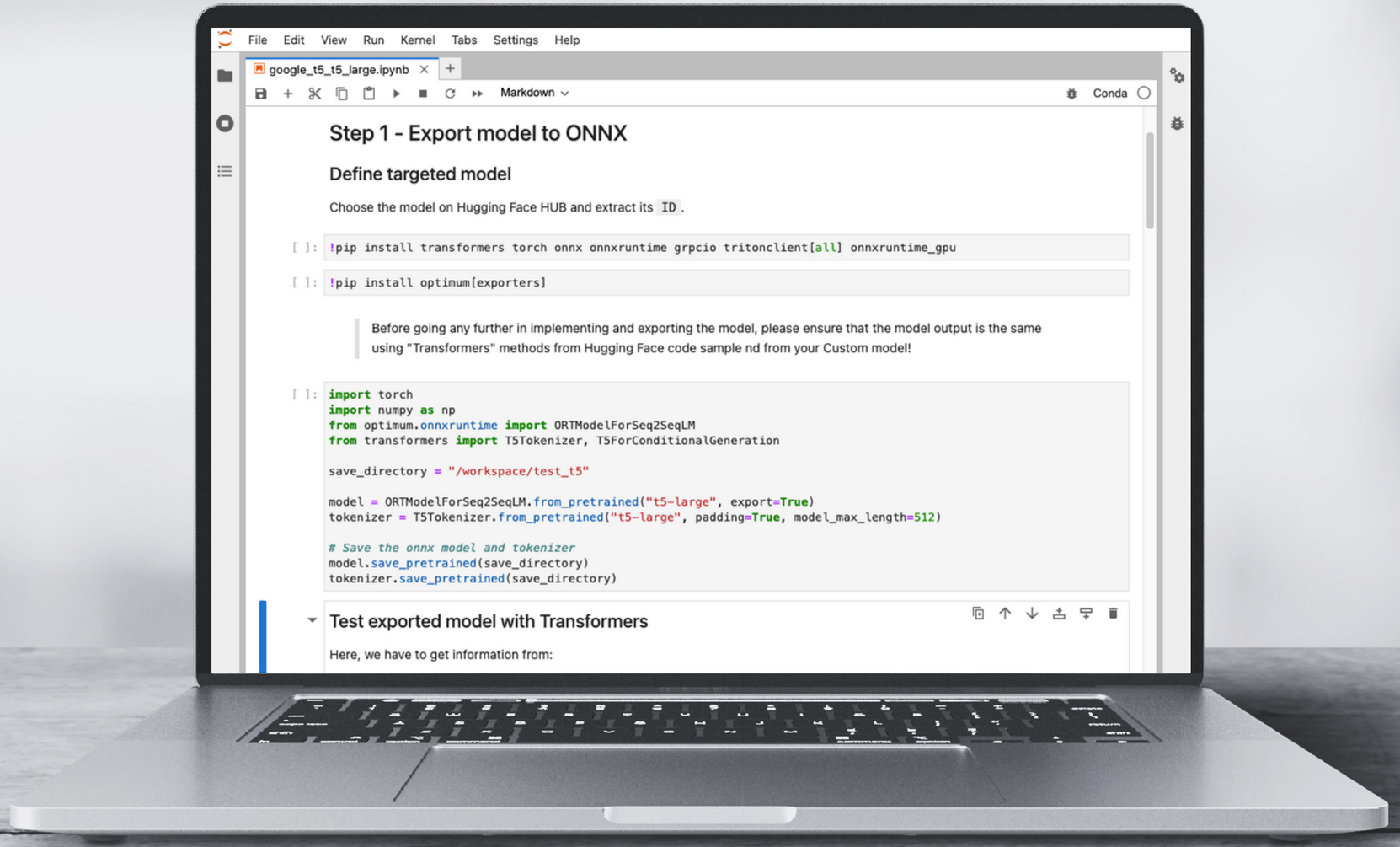
**STOCKER**
les modèles dans un bucket S3

**DÉPLOYER**
le serveur d'inférence dans le cloud




# EXPORTER
## les modèles

# EXPORTER LES MODÈLES AU FORMAT ONNX



# COMMENT DÉPLOYER CES MODÈLES ?

**ACCÉDER**
à un notebook avec GPUs

**TESTER**
le(s) modèle(s) localement

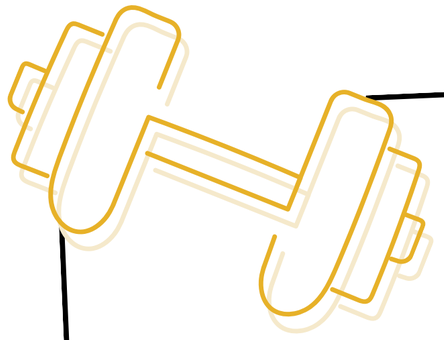
**EXPORTER**
les modèles au format ONNX

**DÉVELOPPER**
les fonctions de pré/post processing

**FORMATER**
le code et le repo de modèles

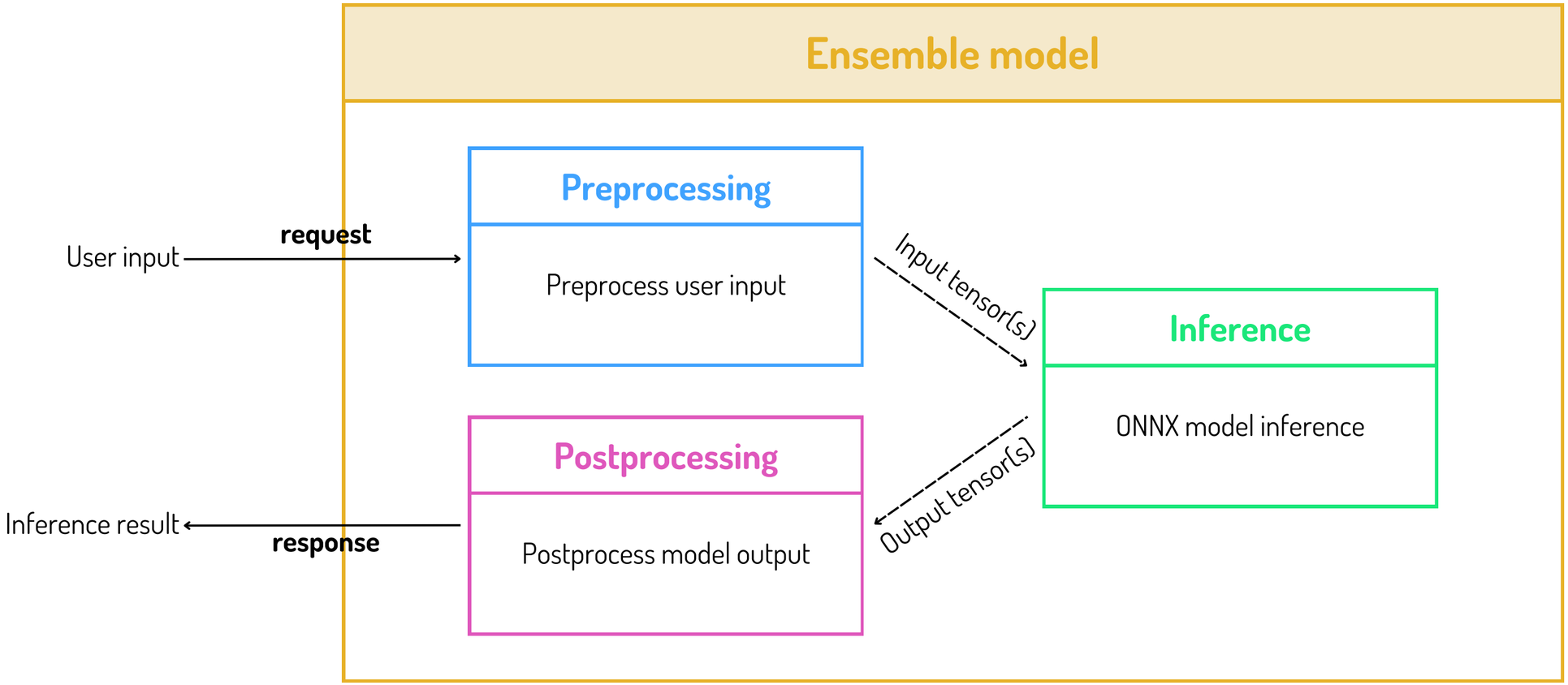
**STOCKER**
les modèles dans un bucket S3

**DÉPLOYER**
le serveur d'inférence dans le cloud



# DÉVELOPPER
## les fonctions pre/post

# DÉVELOPPER LES FONCTIONS DE PRÉ/POST PROCESSING



# COMMENT DÉPLOYER CES MODÈLES ?

**ACCÉDER**
à un notebook avec GPUs

**TESTER**
le(s) modèle(s) localement

**EXPORTER**
les modèles au format ONNX

**DÉVELOPPER**
les fonctions de pré/post processing

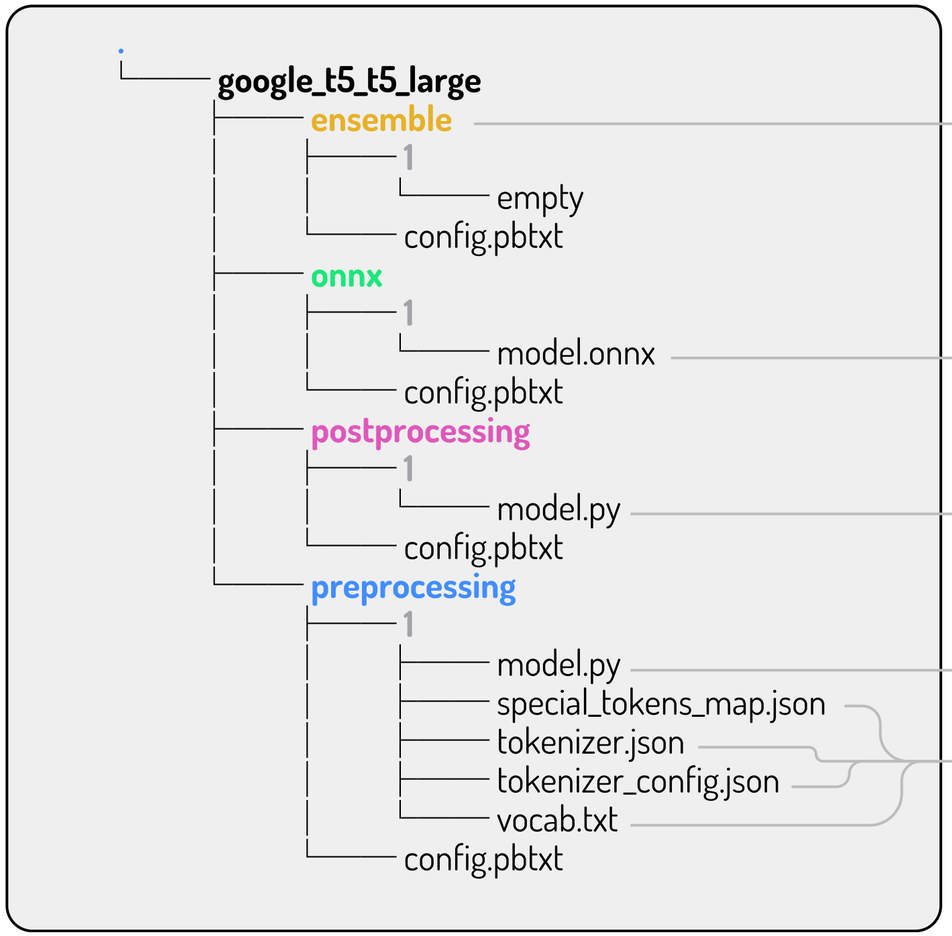
**FORMATER**
le code et le repo de modèles

**STOCKER**
les modèles dans un bucket S3

**DÉPLOYER**
le serveur d'inférence dans le cloud

# FORMATER
## le code et le repo

# FORMATER LE CODE ET LE REPO DE MODÈLES



# COMMENT DÉPLOYER CES MODÈLES ?

**ACCÉDER**
à un notebook avec GPUs

**TESTER**
le(s) modèle(s) localement

**EXPORTER**
les modèles au format ONNX

**DÉVELOPPER**
les fonctions de pré/post processing

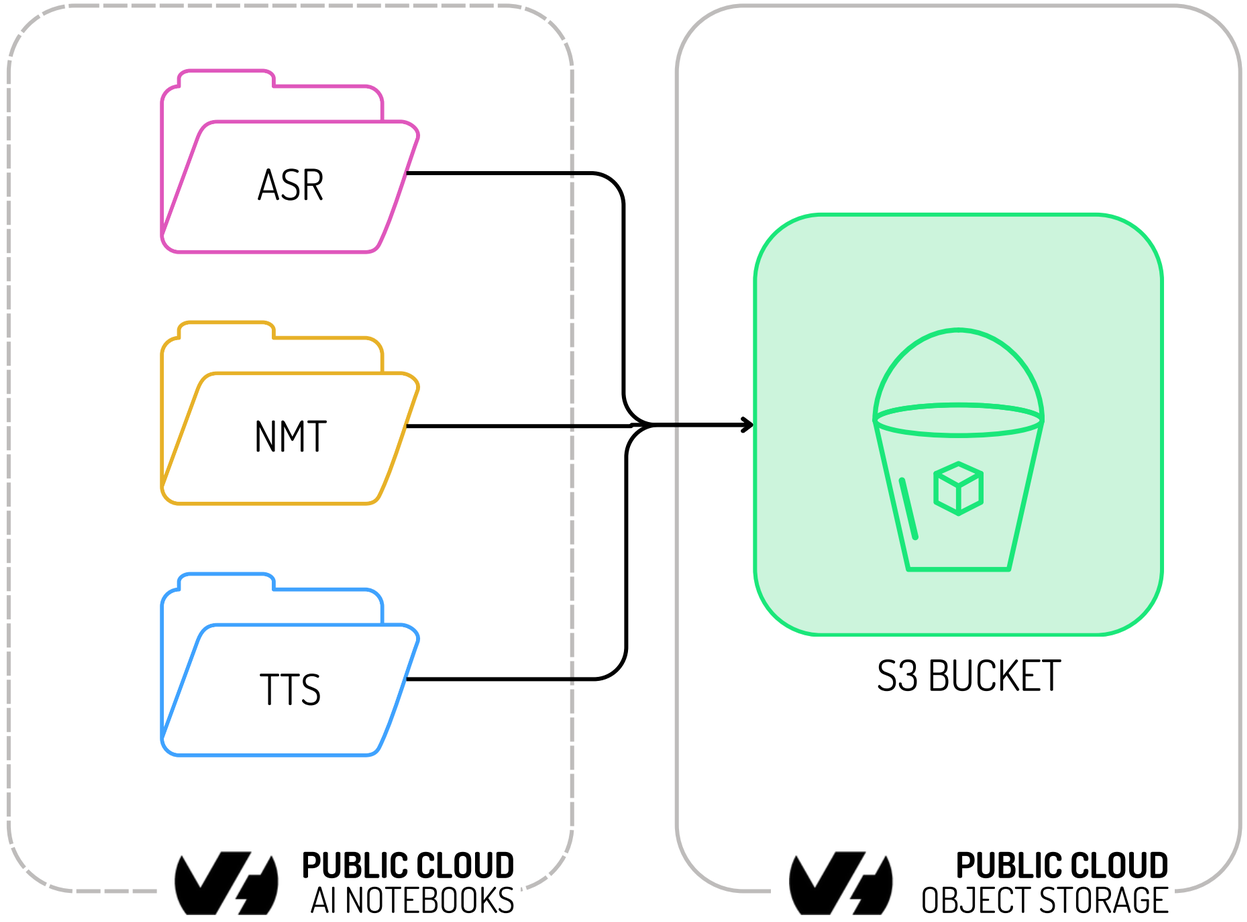
**FORMATER**
le code et le repo de modèles

**STOCKER**
les modèles dans un bucket S3

**DÉPLOYER**
le serveur d'inférence dans le cloud

# STOCKER
## les modèles

# STOCKER LES MODÈLES DANS UN BUCKET S3



# COMMENT DÉPLOYER CES MODÈLES ?

**ACCÉDER**
à un notebook avec GPUs

**TESTER**
le(s) modèle(s) localement

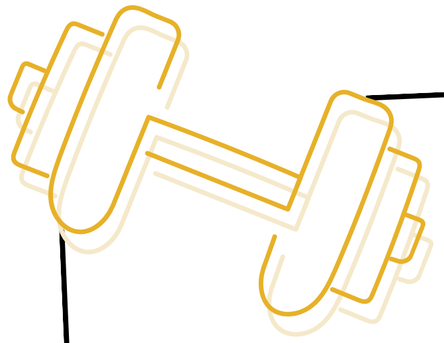
**EXPORTER**
les modèles au format ONNX

**DÉVELOPPER**
les fonctions de pré/post processing

**FORMATER**
le code et le repo de modèles

**STOCKER**
les modèles dans un bucket S3

**DÉPLOYER**
le serveur d'inférence dans le cloud



# DÉPLOYER
## le serveur d'inférence

# DÉPLOYER LE SERVEUR D'INFÉRENCE DANS LE CLOUD


**RESSOURCES DE CALCUL**
- 1 GPU
- V100s

**HAUTE DISPONIBILITÉ**
- API scalable à la volée
- Nombre de réplicas personnalisable

**ACCÈS SÉCURISÉ**
- Model privé
- Token d'accès unique

# OVHcloud - AI DEPLOY

| | |
|---|---|
| 01 CONTENEUR AS A SERVICE | Le client fournit un conteneur Docker via une registry Docker |
| 02 RESSOURCES DE CALCUL | Le conteneur s'exécute dans le cloud sur GPU (ou CPU) |
| 03 MODE DE FACTURATION | Le client est facturé à la minute utilisée |
| 04 DÉPLOIEMENT API/APP | Manière industrielle de déployer des API(s) stateless |
| 05 STRATÉGIE D'AUTOSCALING | Scalable à la volé |

# INFERENCE SERVER AVEC AI DEPLOY

Select the flavor and the number of GPUs

OVHcloud instance

```
ubuntu@nvidiainstance:~$ ovhai app run --name nmt_model \
>               --flavor ai1-1-gpu \
>               --gpu 1 \
>               --default-http-port 8000 \
>               --grpc-port 8001 \
>               -v my_model_repository@S3GRA/google_t5_t5_large/:/data:ro \
>               -v standalone:/workspace:rw \
>               nvcr.io/nvidia/tritonserver:22.01-py3 \
>               -- bash -c 'pip install transformers torch &&
                           tritonserver --model-store /data'
```

# INFERENCE SERVER AVEC AI DEPLOY

Precise HTTP and gRPC port

OVHcloud instance

```
ubuntu@nvidiainstance:~$ ovhai app run --name nmt_model \
>               --flavor ai1-1-gpu \
>               --gpu 1 \
>               --default-http-port 8000 \
>               --grpc-port 8001 \
>               -v my_model_repository@S3GRA/google_t5_t5_large/:/data:ro \
>               -v standalone:/workspace:rw \
>               nvcr.io/nvidia/tritonserver:22.01-py3 \
>               -- bash -c 'pip install transformers torch &&
>                           tritonserver --model-store /data'
```

# INFERENCE SERVER AVEC AI DEPLOY

Attach your S3 buckets with your NMT models

OVHcloud instance

```
ubuntu@nvidiainstance:~$ ovhai app run --name nmt_model \
>               --flavor ai1-1-gpu \
>               --gpu 1 \
>               --default-http-port 8000 \
>               --grpc-port 8001 \
>               -v my_model_repository@S3GRA/google_t5_t5_large/:/data:ro \
>               -v standalone:/workspace:rw \
>               nvcr.io/nvidia/tritonserver:22.01-py3 \
>               -- bash -c 'pip install transformers torch &&
>                           tritonserver --model-store /data'
```

# INFERENCE SERVER AVEC AI DEPLOY

Select Triton Server docker image

OVHcloud instance

```
ubuntu@nvidiainstance:~$ ovhai app run --name nmt_model \
>               --flavor ai1-1-gpu \
>               --gpu 1 \
>               --default-http-port 8000 \
>               --grpc-port 8001 \
>               -v my_model_repository@S3GRA/google_t5_t5_large/:/data:ro \
>               -v standalone:/workspace:rw \
>               nvcr.io/nvidia/tritonserver:22.01-py3 \
>               -- bash -c 'pip install transformers torch &&
>                           tritonserver --model-store /data'
```

# INFERENCE SERVER AVEC AI DEPLOY

Start Triton Inference Server for NMT

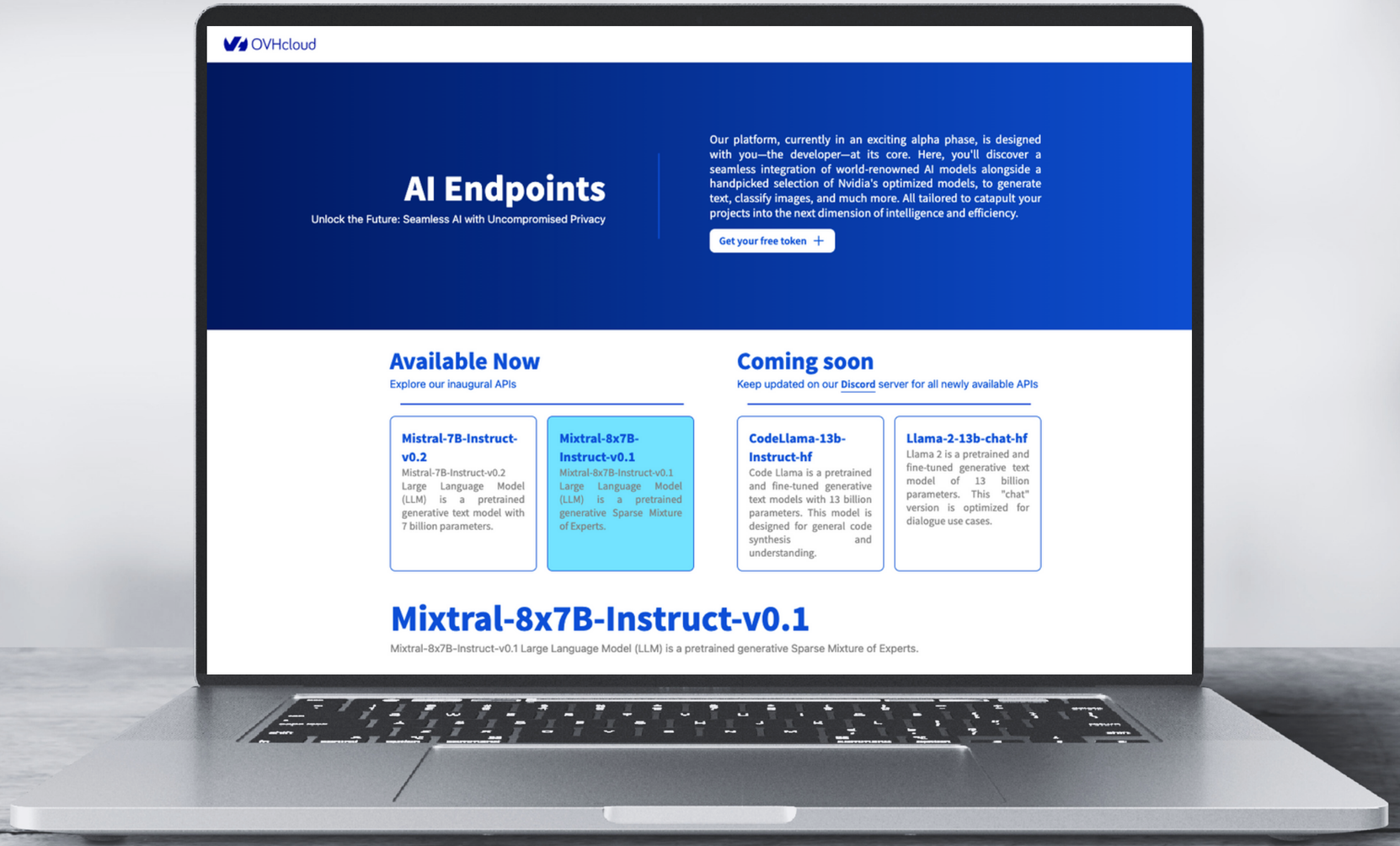
OVHcloud instance

```
ubuntu@nvidiainstance:~$ ovhai app run --name nmt_model \
>               --flavor ai1-1-gpu \
>               --gpu 1 \
>               --default-http-port 8000 \
>               --grpc-port 8001 \
>               -v my_model_repository@S3GRA/google_t5_t5_large/:/data:ro \
>               -v standalone:/workspace:rw \
>               nvcr.io/nvidia/tritonserver:22.01-py3 \
>               -- bash -c 'pip install transformers torch &&
                           tritonserver --model-store /data'
```

# BESOIN D'UNE SOLUTION CLÉ EN MAIN ?

# CHOISIR
## ses AI Endpoints

# OVHcloud - AI ENDPOINTS

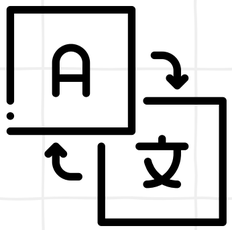


# OVHcloud - AI ENDPOINTS

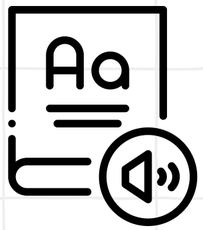
" Choisissez les endpoints API qui correspondent à vos besoins ! "

**ASR**
endpoint

**NMT**
endpoint



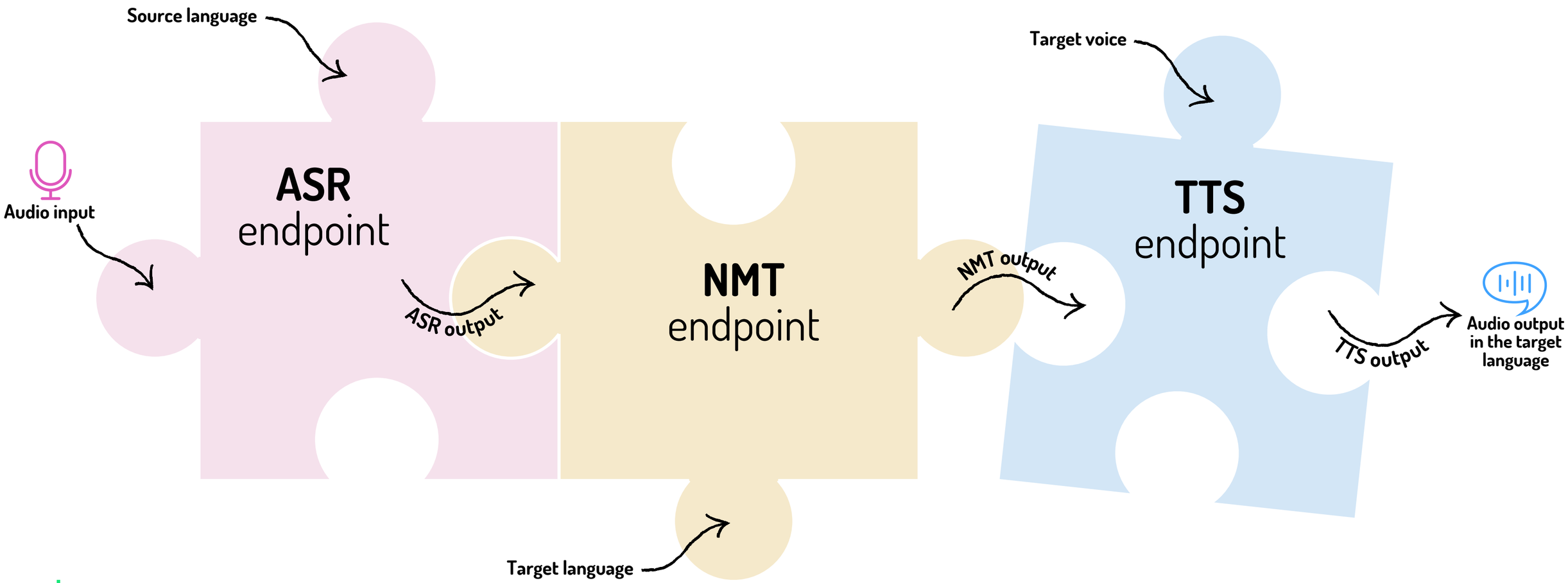

**TTS**
endpoint

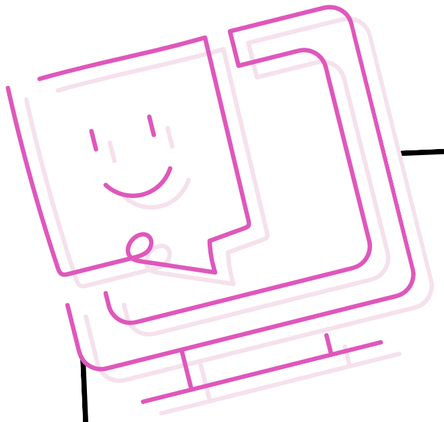


# COMMENT CONNECTER CES AI ENDPOINTS ENTRE EUX ?

# OVHcloud - AI ENDPOINTS

“ Connectez vos AI Endpoints selon vos besoins ! ”





# DÉVELOPPER
## la partie client

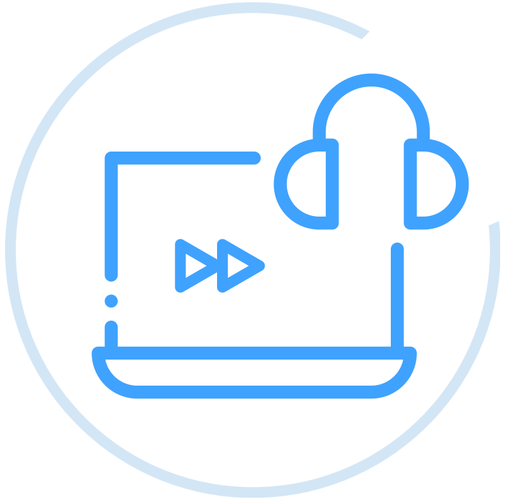
# DÉVELOPPER LA PARTIE CLIENT

- **Entrer** un lien de vidéo YouTube
- **Transcrire** la partie audio de la vidéo en texte
- **Sous-titrer** la vidéo dans n'importe quelle langue
- **Doubler** la voix de le vidéo dans une autre langue
- **Choisir** le genre de la voix de doublage
- **Télécharger** la vidéo résultante

# DÉVELOPPER LA PARTIE CLIENT


**GÉNÉRER**
un fichier SRT de sous-titres

**CONSERVER**
les silences pendant la traduction



**DOUBLER**
l'audio d'une vidéo dans une autre langue

# DÉVELOPPER LA PARTIE CLIENT


**GÉNÉRER**
un fichier SRT de sous-titres

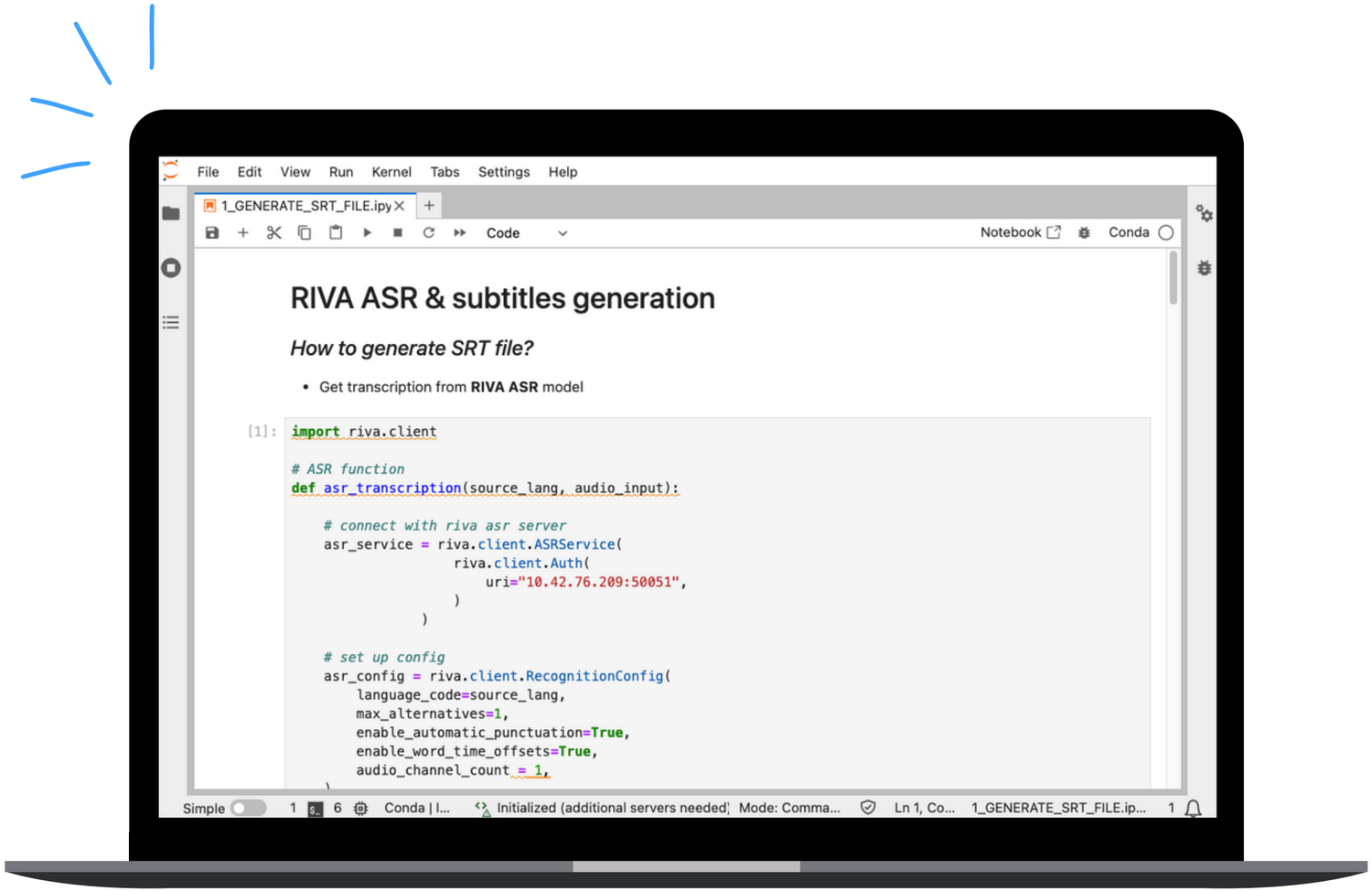


**CONSERVER**
les silences pendant la traduction



**DOUBLER**
l'audio d'une vidéo dans une autre langue

# GÉNÉRER UN FICHIER SRT



# DÉVELOPPER LA PARTIE CLIENT



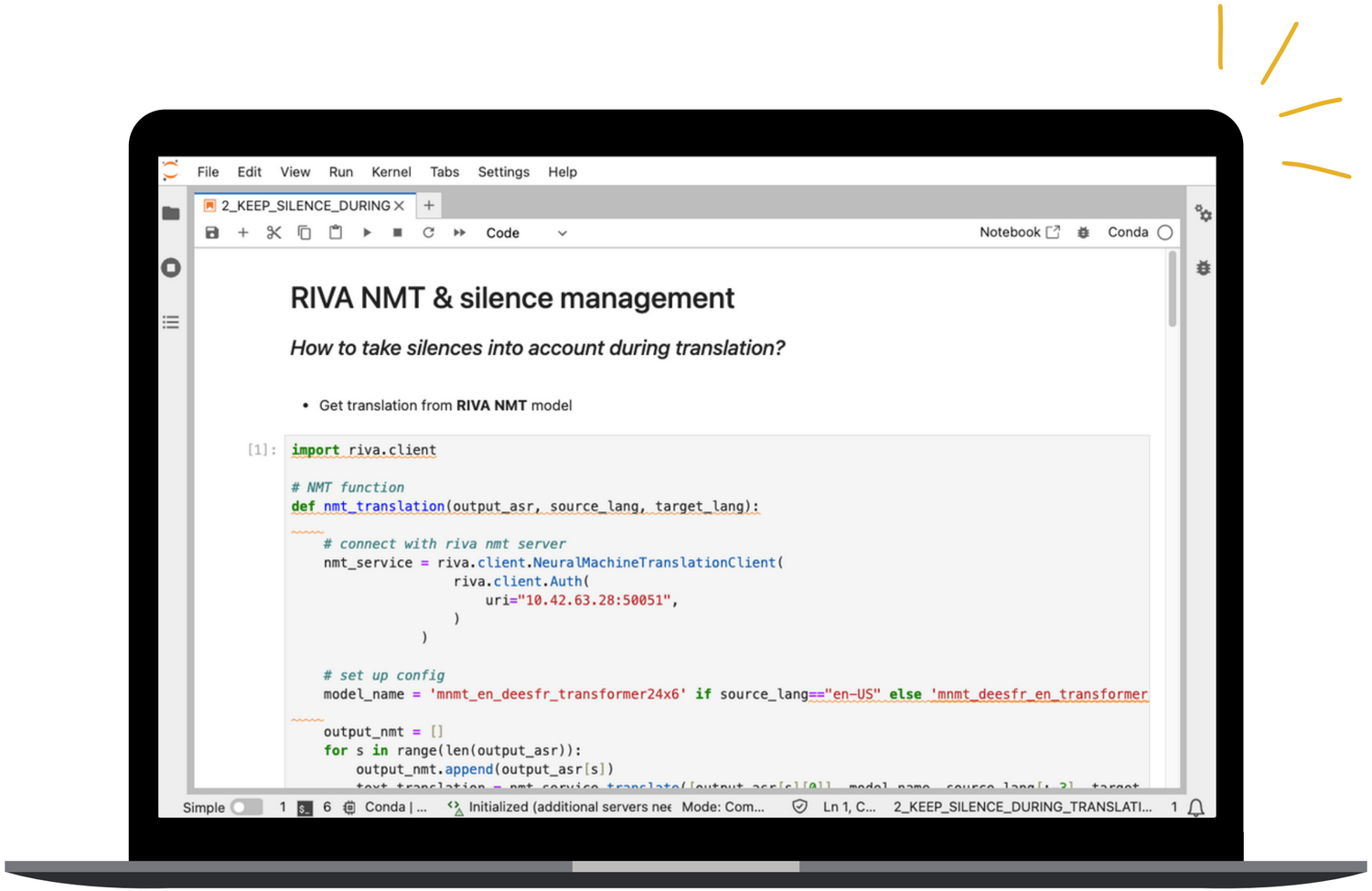

**GÉNÉRER**
un fichier SRT de sous-titres

**CONSERVER**
les silences pendant la traduction



**DOUBLER**
l'audio d'une vidéo dans une autre langue

# CONSERVER LES SILENCES



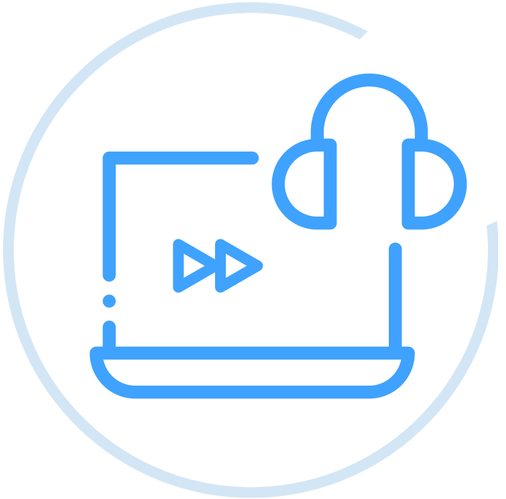
# DÉVELOPPER LA PARTIE CLIENT




**GÉNÉRER**
un fichier SRT de sous-titres

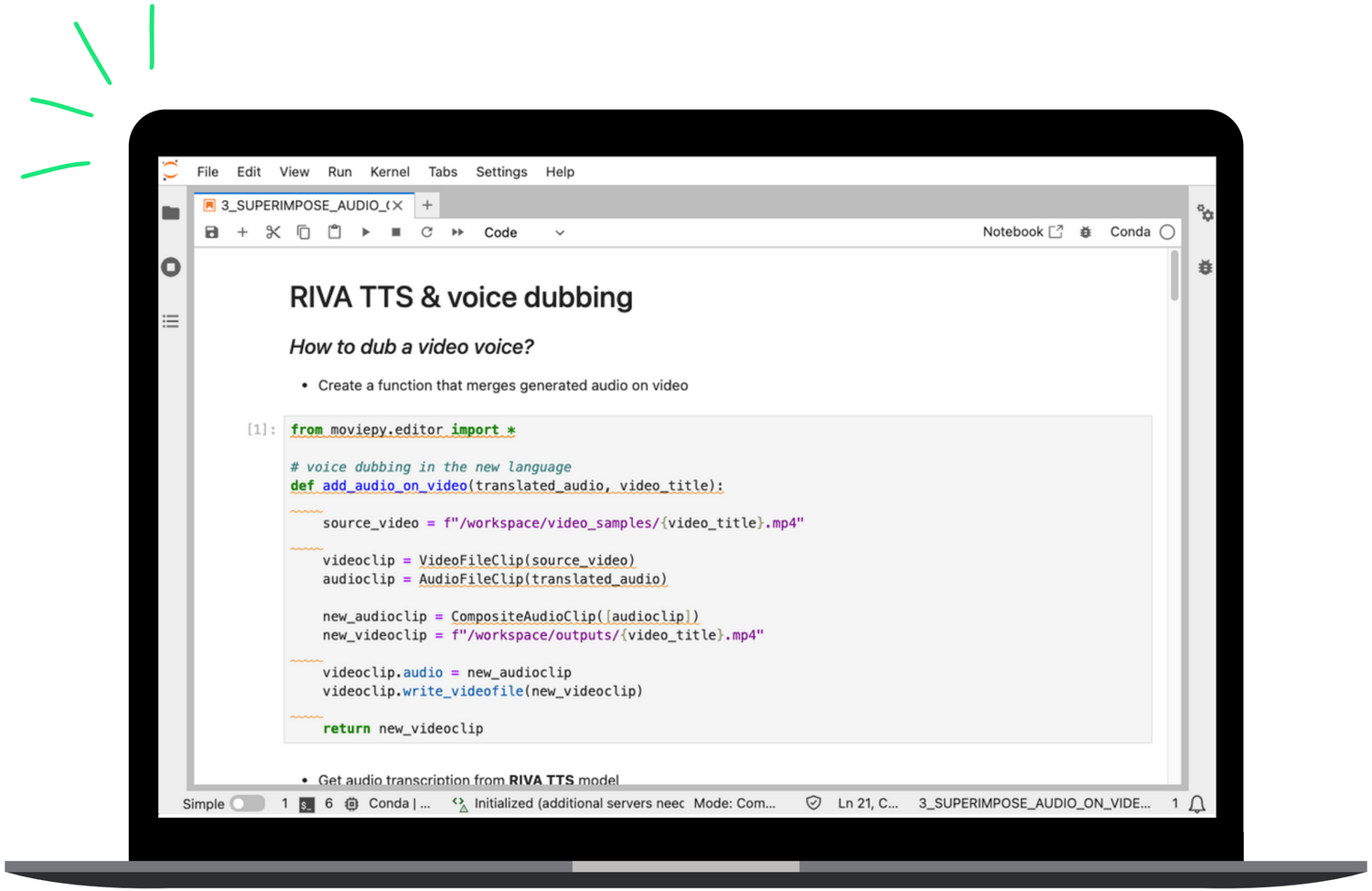


**CONSERVER**
les silences pendant la traduction



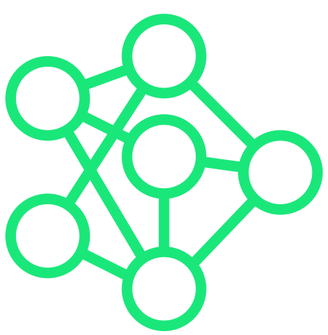
**DOUBLER**
l'audio d'une vidéo dans une autre langue

# DOUBLER L'AUDIO D'UNE VIDÉO

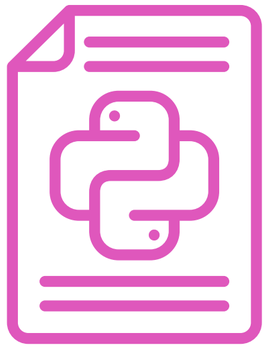


# DÉPLOYER
## l'app end-to-end

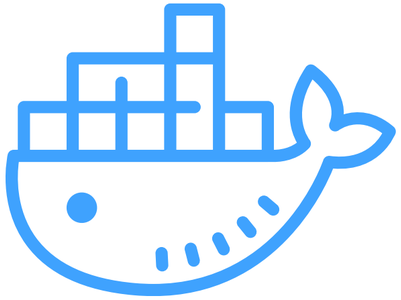
# DÉPLOYER L'APP END-TO-END



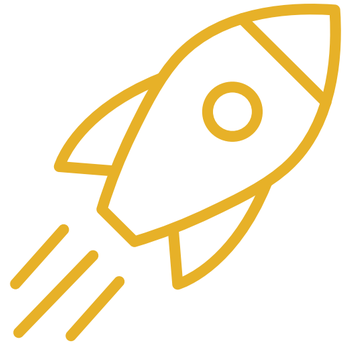
Inference Servers



Custom client code

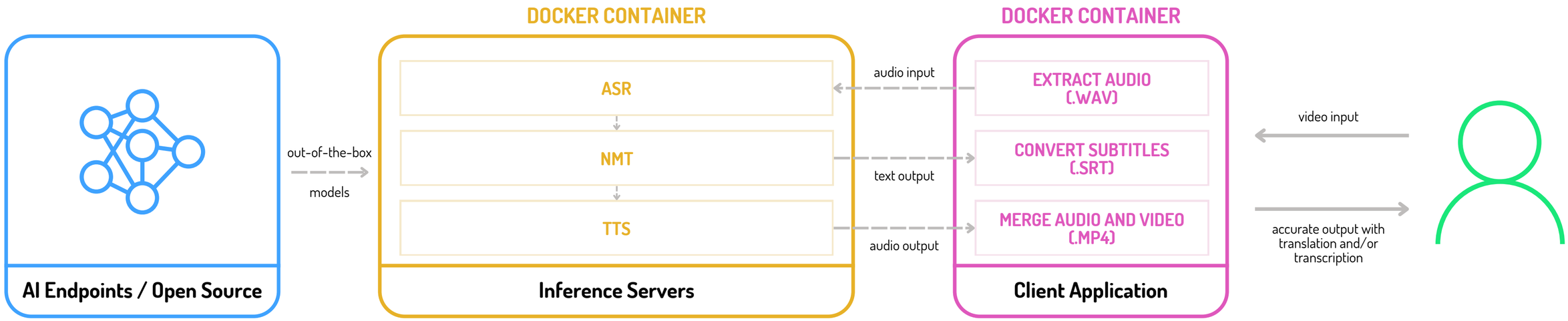


Docker image



AI Deploy app

# DÉPLOYER L'APP END-TO-END




CHOOSE **MODELS** ACCORDING TO YOUR NEEDS

DEPLOY/SELECT **INFERENCE SERVERS** IN THE CLOUD

DEPLOY **CLIENT** TROUGH **GRADIO APP** IN THE CLOUD WITH HIGH AVAILABILITY

# DÉMO
## de l'app

# DÉMO DE L'APP


Multimedia Translator
Powered by OVHcloud
This application lets you transcribe and translate your videos for a global audience using NVIDIA's RIVA models.
Add your video, select the languages, choose the voice and let the magic happen!
Enter the YouTube URL
https://www.youtube.com/watch?v=PF2h-w-FCIc
Translation mode
Subtitles
Voice dubbing
Source language
fr-FR
Target language
en-US
Submit
Translated video
OVHcloud
Ovh Cloud is a global player and the leading European
0:01 / 2:51
Use via API
Built with Gradio

# DÉMO DE L'APP

https://bit.ly/multimedia-translator-devoxx

# CONCLUSION

# CONCLUSION

Le “**multimode multimedia translator**“ en résumé...

**SOLUTION DE TRANSCRIPTION ET TRADUCTION MULTIMÉDIA**

- vidéos traduites et sous-titrées
- vidéos doublées dans une autre langue

**OVHcloud AI ENDPOINTS**

- simple d'utilisation
- gain de temps dans le développement

OVHcloud

**OVHcloud AI DEPLOY**

- solution serverless
- top des GPUs (H100, A100, L4, L40S, V100S)

OVHcloud

**ET CE N'EST PAS FINI...**

- implémentation de la reconnaissance des émotions
- ajout de la transcription en temps réel
- personnalisation de la voix

# OVHcloud AI ENDPOINTS

https://endpoints.ai.cloud.ovh.net/

# MERCI !

# À VOUS DE TESTER !

https://bit.ly/multimedia-translator-devoxx