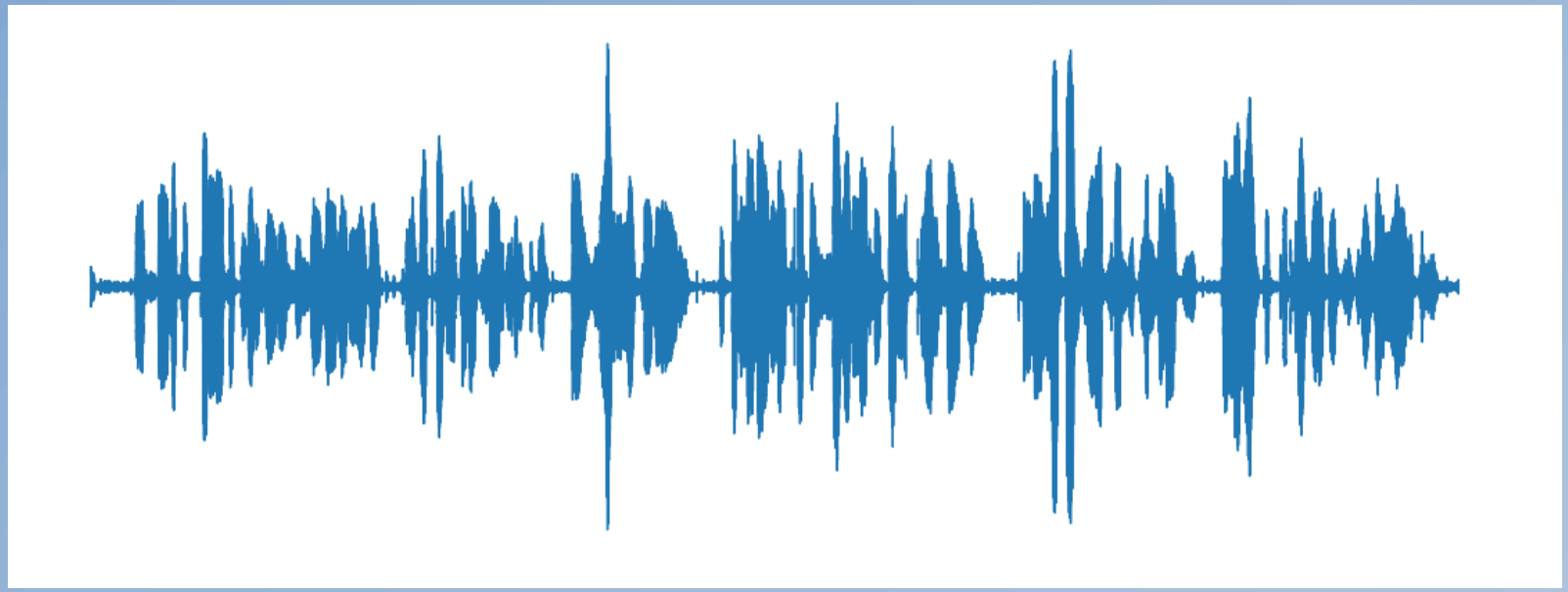
Transcription speech to text
(+ traduction)



Pourquoi ce projet ?

Pas évident de choisir, il y a tant d'outils à tester, et tant d'applications possibles à construire !

L'idée de départ était de réaliser un projet simple mais utile, en travaillant avec un type de données que je n'avais pas encore utilisé pour le machine learning : les données audio.

J'ai d'abord pensé à réaliser une classification audio. C'est intéressant d'un point de vue technique, pour aborder le traitement de la feature audio (preprocessing, feature extraction). Cependant les applications semblaient relativement limitées.

En revanche, un bon algorithme de transcription peut être très utile, dans de nombreuses situations :

- prise de notes,

- envie d'avoir les paroles d'une chanson dans une langue étrangère,

- générer des sous-titres, ou des transcriptions, pour qu'un public malentendant puisse accéder à des ressources audio (émissions, conférences…)

C'est donc notre objectif premier ici : tester les performances des modèles récents (et open-source) en matière de transcription speech-to-text.

En implémentant ces modèles, nous allons cependant vite nous apercevoir… qu'il serait dommage de s'arrêter là !

Les étapes de cette présentation

1) Plan prévisionnel : Jeu de données, modèles, sources

2) Démarche mise en œuvre : Concepts et techniques, comparaison des résultats

3) Démonstration du dashboard, déployé sur le cloud

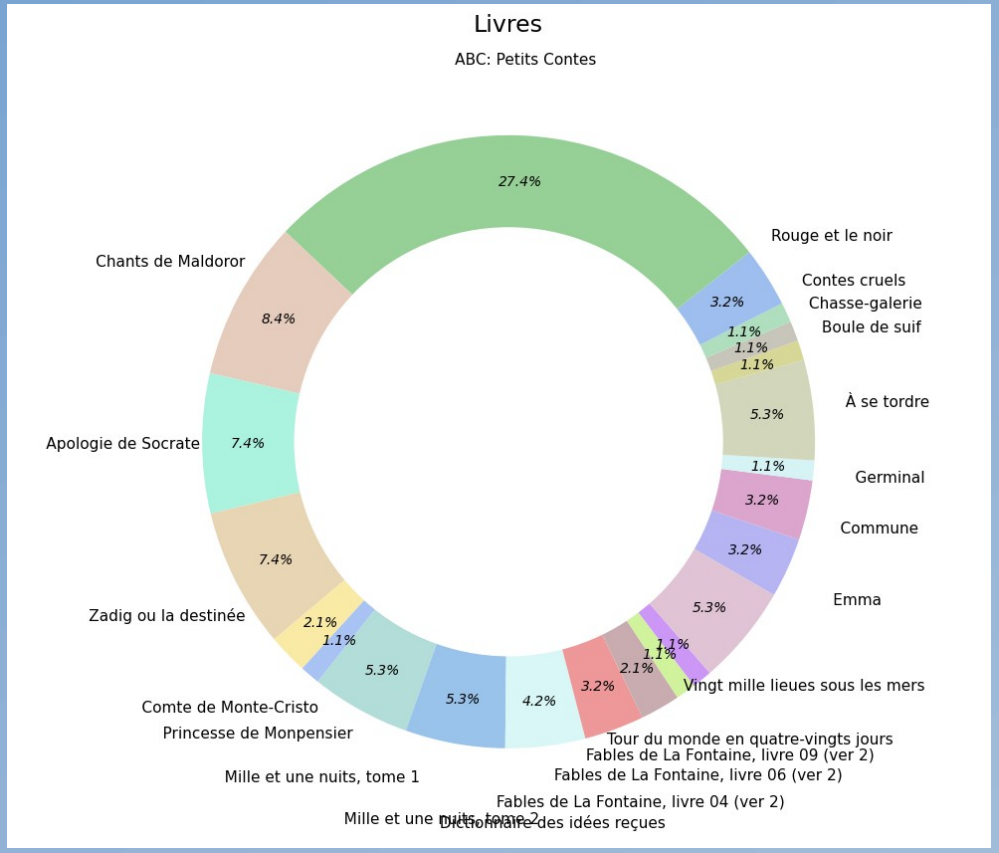
1) Plan prévisionnel : Jeu de données

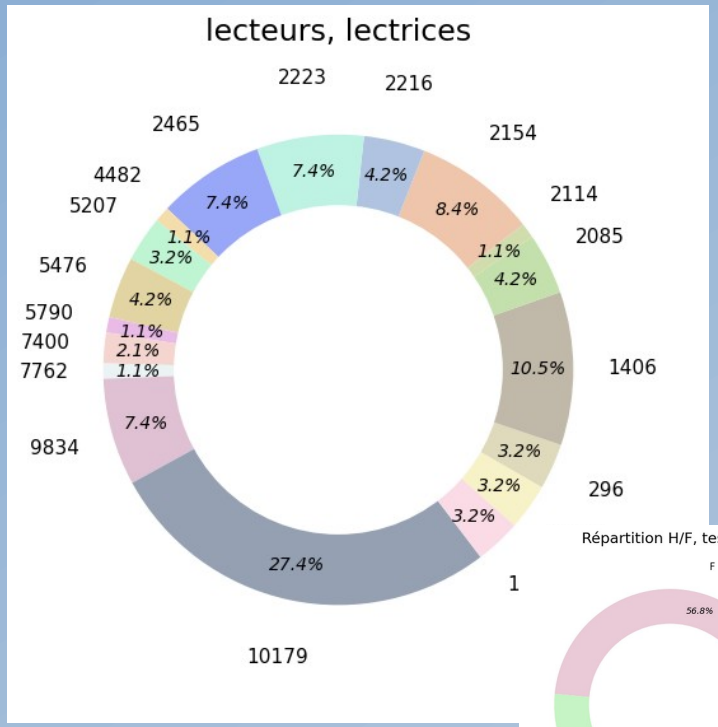
Nous avons besoin d’un dataset contenant des extraits audio et les transcriptions correspondantes. J’ai choisi le Multilingual LibriSpeech (MLS), téléchargeable à l’adresse : https://openslr.org/94

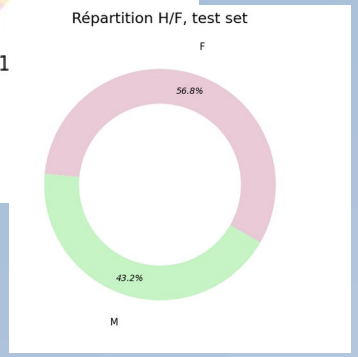
Ce dataset contient des jeux de training, validation et test, pour plusieurs langues. Nos deux modèles sont préentraînés, nous allons donc utiliser une partie des fichiers test, en français.

Il existe deux niveaux de difficulté. Nous allons utiliser le plus difficile : textes littéraires, vocabulaire rare, soutenu ou archaïsant, prononciation parfois mal articulée, ... afin de tester au mieux les capacités de nos modèles.

Informations générales







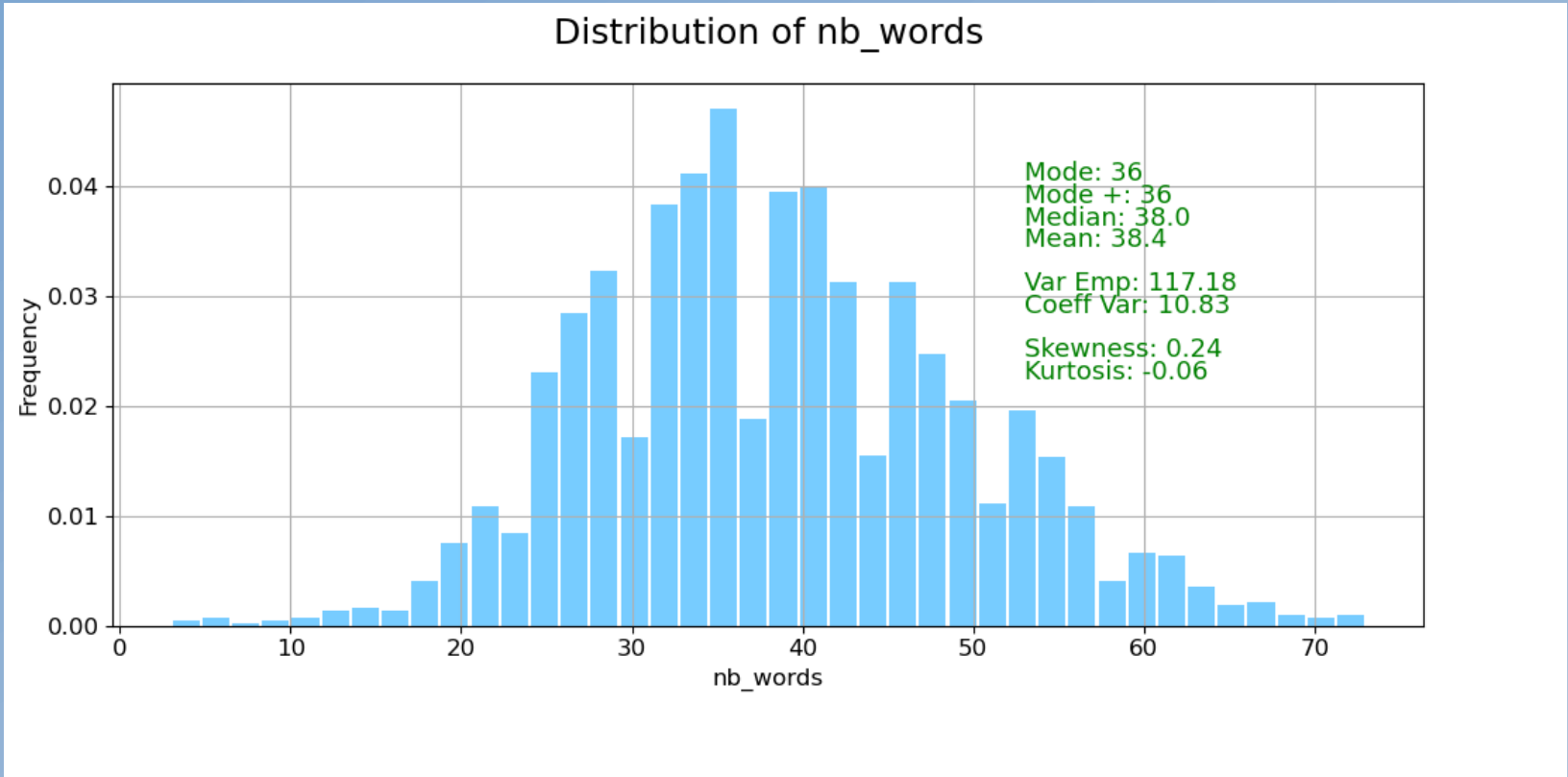
Texte : quelques exemples de transcripts

0 blanqui et millière sortirent également le gouvernement n'osant pas d'abord montrer son mépris de la parole donnée le soir même du octobre avait lieu à la bourse une réunion des officiers de la garde nationale

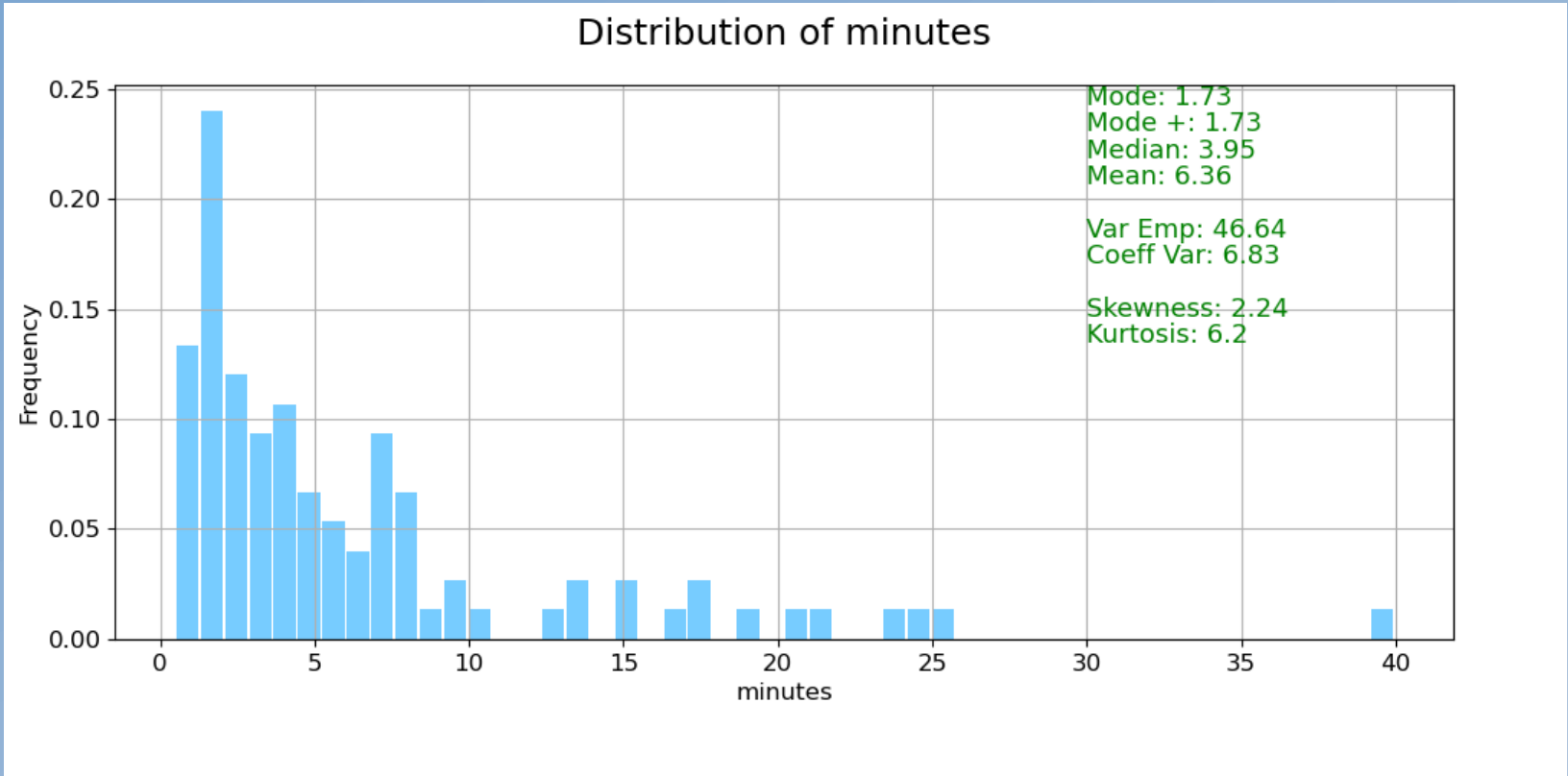
1 il n'est pas si vilain que ça dirent les cygnes et un monsieur cygne avec un magnifique plastron blanc et de beaux pieds vernis déclara qu'il reste parmi nous et dans trois mois je lui donne ma fille en mariage

2 voilà ce qu'il y a de plus difficile à faire entendre à quelques-uns d'entre vous car si je dis que ce serait désobéir au dieu et que par cette raison il m'est impossible de me tenir en repos vous ne me croirez point et prendrez cette réponse pour une plaisanterie

Nombre de mots



Audio : longueur des extraits



Les modèdes : Wave2vec2, notre baseline

Nécessite un tokenizer (spécifique à la langue que l'on souhaite transcrire), ainsi qu'un modèle différent pour la traduction. Wav2vec 2 .0 est donc un modèle très spécialisé, il fait uniquement de la transcription, pour une langue donnée.

La nouvelle méthode : whisper(s)

Beaucoup plus simples à implémenter, mais quelles seront les performances de nos nouveaux modèles, face à une baseline qui a fait ses preuves ?

Wav2vec et whisper appartiennent à la même famille de modèles transformers. Ils ont donc des architectures relativement similaires, encoder-decoder.

Sources : choix de la nouvelle méthode

- *Houston we have a Divergence: A Subgroup Performance Analysis of ASR Models*, by Alkis Koudounas, Flavio Giobergia, 31 mars 2024

La Nasa utilise Whisper ! Source : https://arxiv.org/abs/2404.07226

- *WavLLM: Towards Robust and Adaptive Speech Large Language Model*, by Shujie Hu, 31 mars 2024

L'université chinoise de Hong Kong utilise le bloc encoder de whisper pour intégrer le speech to text à leur projet de LLM. Source : https://arxiv.org/abs/2404.00656

- https://deepgram.com/learn/benchmarking-top-open-source-speech-models

Il ne s'agit pas d'un article de recherche, mais on peut consulter cette page pour une explication de la métrique standard pour la transcription, le WER, et un aperçu des performances du model par rapport à la baseline.

Sources : notebooks et références techniques

https://www.kaggle.com/code/stpeteishii/french-audio-wav2vec2-translation

Projet servant de baseline. Il a été publié en septembre 2023. Pour plus d’informations sur wav2vec2, voire

https://jonathanbgn.com/2021/09/30/illustrated-wav2vec-2.html

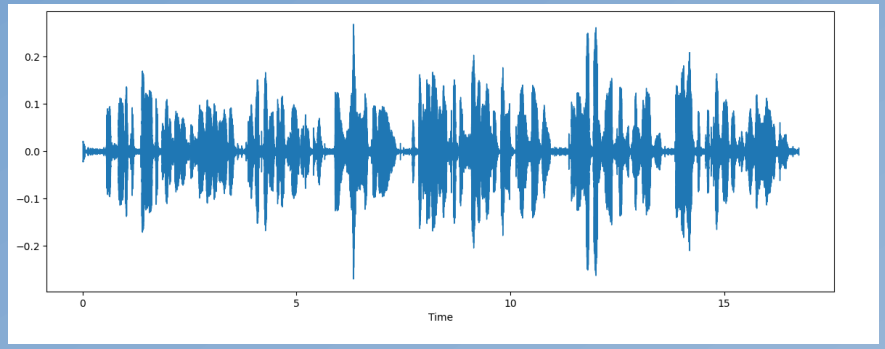
https://medium.com/axinc-ai/whisper-speech-recognition-model-capable-of-recognizing-99-languages-5b5cf0197c16

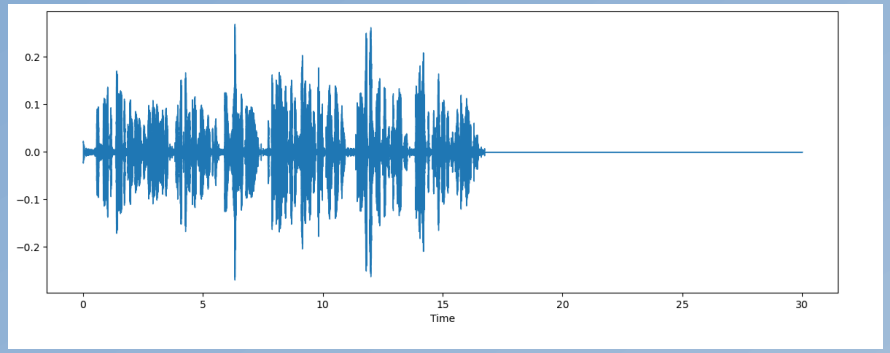
Concepts de la nouvelle méthode

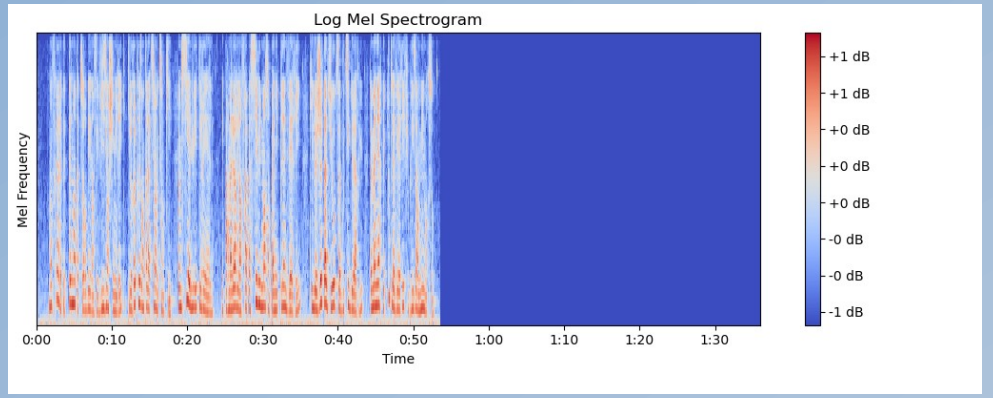
Métrique : https://www.kaggle.com/code/kevinvaishnav/speechtotext

2) Démarche

Après avoir nettoyé nos données (1 doublon), nous pouvons procéder au feature engineering des enregistrements audio. Pour whisper :







Evaluer la qualité des transcriptions obtenues

Métrique : le WER, construit en utilisant la distance de Levenshtein. Exemples :

```
print(_levenshtein_distance('abc', 'abcd')) # 1 transformation nécessaire  -> 1
print(_levenshtein_distance('abc', 'ABC'), '\n') # Sensible à la casse !       -> 3

print(wer('Ils sont partis', 'ils sont parti')) # taux = 2 erreurs / 3 mots         -> 0.66...
print(wer('Ils sont partis', 'i sont pati')) # Compte une erreur max par mot -> 0.66...
```

Résultats pour la transcription

...

	model	size test set	WER moyen	WER std	time predict moyen (s)
4	whisper_large	300	0.070	0.098	74.27
1	whisper_medium	300	0.117	0.208	39.64
2	wav2vec2	300	0.139	0.115	4.05
0	whisper_small	300	0.156	0.140	12.41
3	whisper_base	300	0.263	0.165	4.62
5	whisper_tiny	300	0.383	0.181	2.8

Exemple de résultats pour la traduction (fr → en)

Texte fr d'origine : il n'est pas si vilain que ça dirent les cygnes et un monsieur cygne avec un magnifique plastron blanc et de beaux pieds vernis déclara qu'il reste parmi nous et dans trois mois je lui donne ma fille en mariage

Baseline : It is not a villain that they say the cygnes and a mseu sign with a magnificent plastron blanc et de beaux pieds Vernis declares that he remains among us and in three months I will give him my daughter in marriage

Whisper medium : He's not that mean, the signs say. And a man signs with a magnificent white plaster and beautiful-glued feet, declares, that he stays among us, and in three months I will give him my daughter in marriage.

Conclusion baseline :

Avantages :

- rapide et efficace

Inconvénients :

- ne fonctionne que pour une langue donnée,
- beaucoup plus compliqué à mettre en production pour du multiligue,
- des méthodes + performanes commencent à être développées.

Conclusion nouvelle méthode

Avantages :

- extrêmement simple à implémenter,
- très bonnes performances

Inconvénients :

- temps de prédiction beaucoup plus important pour les meilleurs modèles,
- encore beaucoup d’hallucinations.

URL de l’appli réalisée :

https://whisper-medium.streamlit.app/