Université Hassan II

Faculté des sciences Ben M’sick

Système de transcription audio basé sur un modèle neuronal pré-entraîné

Réalisé par :

- **Issalmou Adaaiche**
- **Oualid Almou**
- **Aya Adadi**

Encadré par :

- **El Habib Ben lahmar**
- **Zakaria Elfakir**

Année universitaire : 2024-2025

Dédicace

Nous tenons à exprimer notre profonde gratitude à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce projet.

Nos remerciements les plus sincères vont à nos parents, pour leur amour constant, leur soutien indéfectible et les nombreux sacrifices qu'ils ont consentis pour nous permettre de poursuivre nos études dans les meilleures conditions.

Nous remercions également nos enseignants pour la qualité de leur encadrement, leur disponibilité et leur engagement tout au long de notre parcours. Leurs conseils avisés et leur exigence bienveillante ont été pour nous une source d'inspiration et de motivation précieuse.

Nous n'oublions pas nos amis, dont le soutien moral, les échanges enrichissants et les moments de convivialité partagés ont apporté équilibre et énergie à notre cheminement.

Enfin, nous remercions toutes les personnes qui, par un mot, un geste ou un encouragement, ont participé à leur manière à l'aboutissement de ce travail.

Remerciement

Nous tenons à exprimer notre profonde gratitude à toutes les personnes qui ont contribué, d'une manière ou d'une autre, à l'aboutissement de ce travail.

Tout d'abord, nous remercions sincèrement nos encadrants, Mr El Habib BENLAHMER et Mr Zakaria ELFAKIR, pour leur disponibilité, leurs conseils précieux, ainsi que leur accompagnement méthodologique et leur soutien constant tout au long de ce projet.

Nous remercions également l'ensemble du corps enseignant de la Faculté des Sciences Ben M'Sik pour la qualité de l'enseignement reçu durant toute notre formation.

Un grand merci à nos camarades de promotion pour leur entraide, leurs échanges enrichissants et l'esprit de collaboration qui a régné durant toutes ces années.

Enfin, nous remercions nos familles pour leur amour, leur patience et leur confiance en nous, ainsi que toutes les personnes qui nous ont apporté de l'aide, des encouragements ou de la motivation dans les moments les plus difficiles.

Abstract

This project explores the use of automatic speech transcription technologies powered by artificial intelligence. We implemented a complete pipeline for processing and transcribing audio recordings using OpenAI's Whisper model.

The workflow includes an audio preprocessing stage to enhance input quality, followed by automatic transcription and performance evaluation using the Word Error Rate (WER) metric. Experiments were conducted on an annotated dataset containing audio files with reference transcriptions.

The results demonstrate satisfactory model performance, with an average WER indicating strong transcription capabilities across varied conditions.

This work highlights the effectiveness and relevance of open-source solutions for speech recognition tasks and paves the way for continuous improvements and real-world applications across multiple domains.

Résume

Ce projet s'inscrit dans le cadre de l'exploration des technologies de transcription automatique de la parole à l'aide de modèles d'intelligence artificielle. Nous avons mis en œuvre une solution complète de traitement et de transcription d'enregistrements audio en utilisant le modèle Whisper développé par OpenAI. Le processus inclut une phase de prétraitement des signaux sonores afin d'améliorer la qualité des données d'entrée, suivie d'une transcription automatique, puis d'une évaluation des performances à l'aide de la métrique Word Error Rate (WER). Les expérimentations ont été menées sur un corpus annoté contenant des enregistrements audio accompagnés de leurs transcriptions de référence. Les résultats obtenus montrent une performance satisfaisante du modèle, avec une moyenne de WER indiquant une bonne capacité de transcription dans des conditions variées. Ce travail met en évidence l'efficacité et la pertinence des solutions open source pour les tâches de reconnaissance vocale, et ouvre la voie à des perspectives d'amélioration continue et d'applications concrètes dans divers domaines.

Table de matière

Table de figures

Introduction générale

La transcription automatique de la parole est une technologie clé qui permet de convertir des données audio en texte écrit de manière rapide et efficace. Cette technologie trouve des applications dans de nombreux domaines, tels que la création de sous-titres, l'accessibilité pour les personnes malentendantes, l'analyse de contenu audio, ou encore la documentation d'entretiens et de réunions. Avec les progrès récents de l'intelligence artificielle, en particulier les modèles de reconnaissance vocale basés sur l'apprentissage profond, la précision et la robustesse des systèmes de transcription se sont considérablement améliorées.

Dans ce contexte, le modèle Whisper d'OpenAI constitue une avancée importante, offrant une solution open source capable de traiter divers types d'enregistrements audio avec une bonne qualité de transcription, même en présence de bruits de fond ou d'accents variés. Notre projet s'inscrit dans cette dynamique : il vise à mettre en œuvre un pipeline complet intégrant le prétraitement des fichiers audio, la transcription automatique via Whisper, puis l'évaluation des résultats à l'aide d'indicateurs précis comme le Word Error Rate (WER).

L'objectif principal est d'évaluer la performance du modèle sur un corpus spécifique, d'analyser les facteurs influençant la qualité des transcriptions, et d'explorer les possibilités d'optimisation du processus. Ce travail permet également de mieux comprendre les avantages et les limites des technologies actuelles, tout en posant les bases pour des applications futures dans des domaines variés.

Chapitre 1 : Contexte du projet

1. Introduction

La transformation numérique et la croissance exponentielle des contenus audio rendent indispensable le développement d'outils efficaces de transcription automatique. Qu'il s'agisse d'enregistrements professionnels, de podcasts, d'interviews ou d'appels téléphoniques, la conversion de la parole en texte permet d'améliorer l'accessibilité, la recherche d'informations et l'analyse de données. Face à ces enjeux, les modèles de reconnaissance vocale basés sur l'intelligence artificielle ont connu une évolution rapide, offrant aujourd'hui des performances proches de celles des humains dans certaines conditions.

2. Problématique

Malgré les progrès technologiques majeurs réalisés ces dernières années dans le domaine de la reconnaissance vocale, plusieurs défis techniques et pratiques continuent de limiter la performance et la généralisation des systèmes de transcription automatique. Parmi ces défis, la variabilité des voix est l'un des plus complexes à gérer. En effet, les systèmes doivent être capables de comprendre une grande diversité de locuteurs, incluant différentes tonalités, vitesses d'élocution, accents régionaux, et variations individuelles de prononciation. Cette diversité entraîne souvent des erreurs de reconnaissance, en particulier lorsque le modèle n'a pas été suffisamment exposé à ces variations lors de son entraînement.

La qualité des enregistrements audio constitue un autre facteur déterminant. Les enregistrements réalisés dans des environnements non contrôlés souffrent fréquemment de bruits de fond, d'échos, ou de niveaux sonores faibles. Ces perturbations altèrent la clarté du signal vocal et rendent la tâche de transcription plus ardue, augmentant le taux d'erreurs. De plus, la présence de bruits parasites comme des conversations parallèles, des sons mécaniques ou des interférences acoustiques complique la séparation de la voix principale, ce qui nécessite souvent des traitements préalables complexes.

Face à ces enjeux, il est donc crucial de se tourner vers des solutions **open source**, qui offrent une plus grande liberté d'usage, une adaptabilité plus fine.

Le développement et l'évaluation de tels outils permettent d'élargir l'accès aux technologies de reconnaissance vocale, favorisant ainsi leur adoption dans des contextes variés, allant de la recherche scientifique à l'aide aux personnes en situation de handicap, ou encore à l'analyse automatique de contenus audiovisuels.

3. Objectifs

Ce projet vise à explorer, concevoir, mettre en œuvre et évaluer une solution complète de transcription automatique de la parole, s'appuyant sur des technologies modernes d'intelligence artificielle. Les objectifs se répartissent en deux catégories : **techniques** et **fonctionnels**

- Objectifs techniques :
  - Mettre en œuvre un pipeline de transcription automatique basé sur le modèle open source Whisper d'OpenAI, intégrant l'ensemble des étapes nécessaires : prétraitement, transcription et évaluation.
  - Appliquer des techniques de traitement audio (filtrage, normalisation, découpe du silence) pour améliorer la qualité des enregistrements en entrée.
  - Évaluer la performance du système à l'aide de métriques objectives, notamment le Word Error Rate (WER), afin de mesurer l'écart entre les transcriptions générées et les transcriptions de référence.
- Objectifs fonctionnels :
  - Faciliter l'accès à la transcription automatique pour les utilisateurs non spécialistes : Offrir une solution simple d'utilisation, sans nécessiter de compétences techniques avancées, pour permettre à des enseignants, chercheurs, journalistes ou étudiants de convertir leurs enregistrements audio en texte.
  - Améliorer l'accessibilité de l'information pour les publics spécifiques : Permettre aux personnes sourdes ou malentendantes d'accéder plus facilement aux contenus oraux en fournissant une transcription fidèle et automatisée.
  - Gagner du temps dans la prise de notes ou la rédaction de comptes rendus : Réduire le travail manuel de retranscription dans les contextes professionnels (réunions, interviews, conférences) grâce à un système automatisé rapide et efficace.

4. Conclusion

Dans ce contexte, le projet permet d'évaluer l'efficacité d'une solution open source moderne pour la transcription automatique, tout en identifiant ses forces et ses limites. Il contribue ainsi à enrichir les connaissances dans ce domaine et à préparer le terrain pour des développements futurs visant à améliorer la qualité et la flexibilité des systèmes de reconnaissance vocale.

Chapitre 2 : Benchmarking

1. Introduction

Le benchmarking constitue une étape essentielle pour situer notre travail dans le paysage actuel des solutions de transcription automatique open source. Il permet de comparer les principaux modèles existants selon leurs performances, leur robustesse et leur facilité d'utilisation. Cette analyse comparative éclaire le choix du modèle Whisper, en montrant ses avantages par rapport aux autres approches populaires dans le domaine.

2. Modèles existants

Pour mieux situer notre travail et justifier le choix du modèle Whisper, il est important d'examiner les principaux modèles de transcription automatique disponibles, en analysant leurs caractéristiques, leurs performances et leurs limites.

- **Mozilla DeepSpeech :**
  Basé sur des réseaux neuronaux récurrents, DeepSpeech est un modèle open source inspiré des travaux de Baidu. Il offre une architecture simple et une bonne performance sur des données propres. Toutefois, il nécessite un entraînement spécifique et un ajustement selon les corpus, ce qui peut limiter son usage direct sur des données variées.
- **Kaldi :**
  Plus qu'un simple modèle, Kaldi est une boîte à outils complète pour la reconnaissance vocale. Il repose sur des modèles acoustiques traditionnels combinés à des techniques modernes de deep learning. Kaldi est très performant, mais sa complexité et ses besoins en expertise le destinent principalement à des utilisateurs avancés
- **Wav2Vec 2.0 (Facebook AI) :**
  Ce modèle utilise une approche de self-supervised learning sur de grandes quantités de données non annotées, ce qui améliore grandement la robustesse et la capacité à s'adapter à différents domaines. Il est performant mais peut nécessiter une étape de fine-tuning spécifique pour des résultats optimaux.
- **OpenAI Whisper :**
  Whisper est un modèle récent, entraîné sur un vaste corpus multilingue comprenant des données avec bruit et accents variés. Il combine robustesse,

polyvalence et simplicité d'utilisation, ce qui en fait une solution attractive pour des projets variés, notamment dans des environnements bruyants ou multi-langues.

3. Comparaison des modèles

Modèle	Robustesse bruit	Polyvalence linguistique	Facilité d'utilisation
DeepSpeech	Moyenne	Limité	Facile
Kaldi	Élevée	Moyenne	Complexe
Wav2Vec 2.0	Élevée	Bonne	Moyenne
Whisper	Très élevée	Très élevée	Très facile

4. Justification du choix de Whisper

Le choix du modèle Whisper pour ce projet est motivé par plusieurs facteurs clés :

- **Robustesse et précision** : Whisper affiche des performances parmi les meilleures, avec un faible taux d'erreur même dans des conditions acoustiques difficiles.
- **Support multilingue** : contrairement à certains modèles focalisés sur l'anglais, Whisper gère plusieurs langues et accents, ce qui élargit son champ d'application.
- **Facilité d'intégration** : son API simple permet un déploiement rapide, sans nécessiter de compétences avancées en machine learning ou de longues phases de réglage.
- **Open source et gratuit** : il permet de travailler en local, sans dépendance à un service externe, garantissant la confidentialité des données.
- **Entraînement sur données variées** : le corpus massif utilisé pour son entraînement inclut des enregistrements en situation réelle, ce qui améliore sa capacité à gérer les enregistrements bruités.

Ces avantages font de Whisper un modèle particulièrement adapté à notre projet, combinant performance, flexibilité et accessibilité.

5. Conclusion

En résumé, notre analyse comparative des modèles de transcription automatique révèle que le modèle **Whisper** d'OpenAI se distingue par son équilibre entre performance, robustesse multilingue et accessibilité. Contrairement à d'autres architectures qui nécessitent des ajustements complexes ou un entraînement sur mesure, Whisper offre une solution clé en main, efficace même dans des environnements bruités ou avec des accents variés. Le choix du modèle **Whisper medium** a ainsi été motivé par sa capacité à produire des transcriptions de qualité tout en restant exploitable dans un cadre académique et open source.

Chapitre 3 : Méthodologie

1. Introduction

Afin d'atteindre les objectifs définis dans ce projet, une méthodologie rigoureuse a été adoptée, articulée autour de plusieurs étapes clés. Cette démarche comprend la préparation des données audio, l'application de techniques de prétraitement pour améliorer la qualité des enregistrements, l'utilisation du modèle Whisper pour la transcription automatique, ainsi que l'évaluation des performances obtenues. Chaque étape a été conçue de manière à garantir la fiabilité des résultats et à permettre une analyse approfondie des facteurs influençant la précision des transcriptions.

2. Architecture de projet

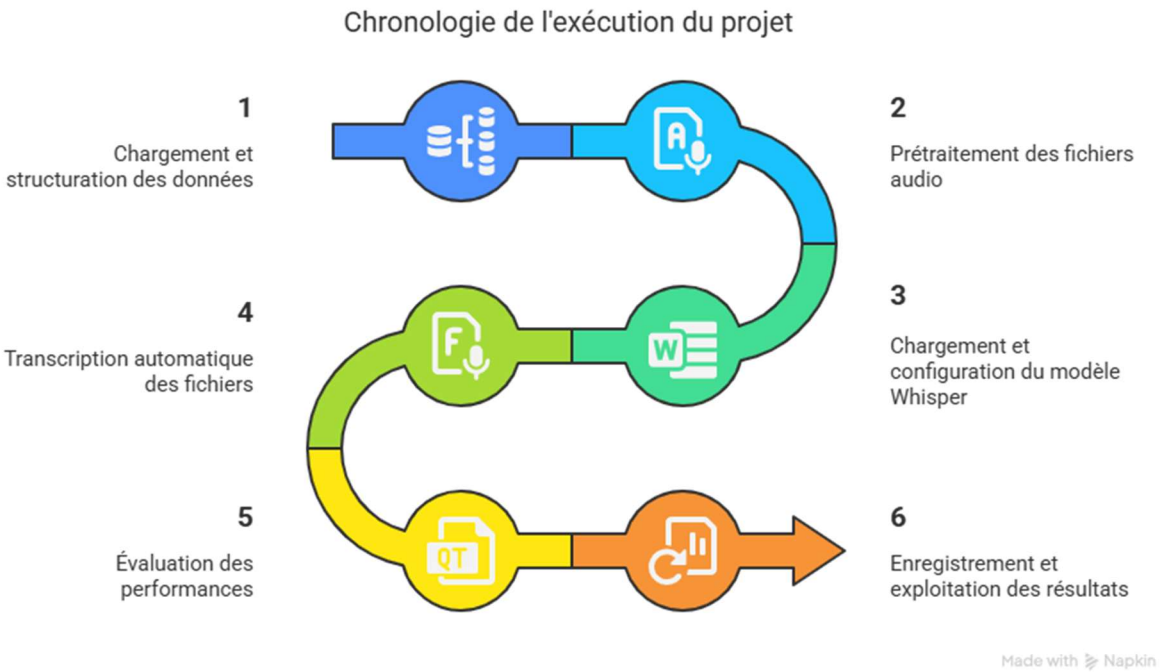

Figure 1: architecture de projet

3. Chargement des données

Les données utilisées dans ce projet proviennent du **dataset *cv-other-test***, une sous-partie du corpus Common Voice, développée par Mozilla, qui contient des enregistrements vocaux accompagnés de leurs transcriptions. Cette base de données est particulièrement utile pour l'évaluation de systèmes de

reconnaissance vocale, car elle offre une grande diversité de voix, d'accents et de conditions d'enregistrement.

Pour faciliter l'exploitation de ces données, un fichier CSV a été généré et stocké dans **Google Drive**. Ce fichier a été importé dans l'environnement **Google Colab** à l'aide de la bibliothèque **Pandas**.

Le fichier contient les colonnes suivantes :

- filename : le nom du fichier audio
- text : la transcription de référence associée (vérité terrain)
- full_path : le chemin absolu vers chaque fichier audio dans le système de fichiers.

4. Prétraitement audio

Afin d'optimiser la qualité des transcriptions produites par le modèle Whisper, un pipeline de prétraitement audio a été mis en place.

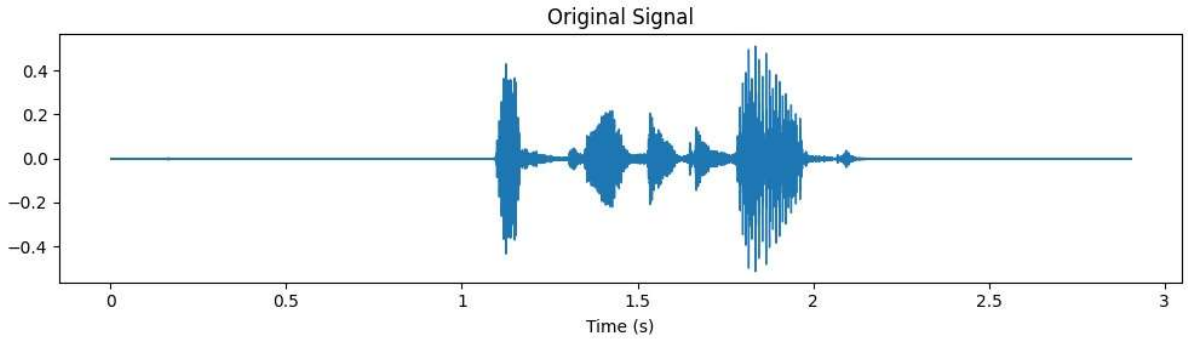

Figure 2: signal original

Ce pipeline vise à nettoyer et normaliser les fichiers audio du dataset *cv-other-test* avant leur transcription. Voici les différentes étapes appliquées :

a) Conversion du format

Objectif : Uniformiser les fichiers dans un format lisible et non destructif (.wav PCM linéaire).

Les fichiers audio peuvent exister dans différents formats : MP3, WAV, M4A, OGG, etc. Certains de ces formats utilisent des compressions avec pertes qui dégradent la qualité sonore, tandis que d'autres ne sont pas compatibles avec certaines bibliothèques utilisées pour l'analyse.

Pour standardiser les données, tous les fichiers ont été convertis vers un format non destructif (comme WAV) à l'aide des outils **pydub** et **ffmpeg**. Cette conversion garantit que tous les fichiers sont lisibles et exploitables par le modèle, sans perte d'information sonore, ce qui est essentiel pour maintenir la précision de la transcription.

b) Filtrage passe-haut (High-pass filter)

Objectif : Éliminer les basses fréquences indésirables telles que les grondements, souffles, ou vibrations basses.

Le filtrage passe-haut est une technique de traitement du signal qui permet de supprimer les fréquences situées en dessous d'un certain seuil, appelé fréquence de coupure. Dans le cadre de ce projet, un filtre de Butterworth d'ordre 5 a été utilisé avec une coupure à 100 Hz. Ce type de filtre est connu pour sa réponse progressive, sans oscillation, ce qui évite de déformer le signal vocal. Le but de cette étape est d'éliminer les basses fréquences non utiles telles que les bruits de fond, les vibrations, ou les grondements dus à l'environnement d'enregistrement. Ces bruits parasites peuvent interférer avec la reconnaissance vocale. En les supprimant, on améliore le rapport signal/bruit et on facilite l'analyse par le modèle.

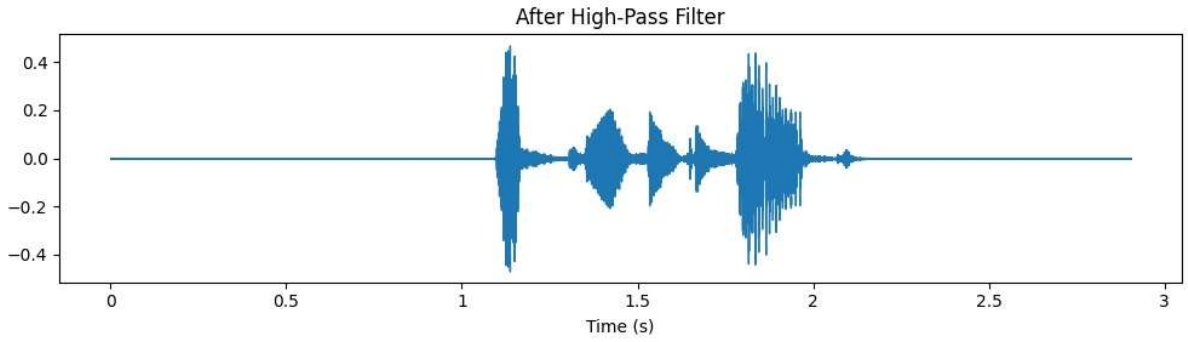

Figure 3: Signal après filtrage passe-haut

c) Rééchantillonnage à 16 kHz

Objectif : Adapter le signal audio à la fréquence d'échantillonnage attendue par le modèle Whisper.

Le ré-échantillonnage est une opération qui consiste à modifier la fréquence d'échantillonnage d'un signal audio, c'est-à-dire le nombre d'échantillons pris par seconde pour représenter le son. Dans ce projet, tous les fichiers ont été convertis en un taux de 16 000 Hz. Cette fréquence est particulièrement adaptée aux

modèles de transcription automatique comme Whisper, qui ont été entraînés sur des corpus audios normalisés à cette valeur. Elle est suffisamment élevée pour capturer l'essentiel des composantes vocales sans enregistrer de fréquences trop élevées qui ne contiennent pas d'informations pertinentes pour la reconnaissance de la parole. Cette étape garantit ainsi l'alignement avec les spécifications du modèle et améliore la cohérence globale des données traitées.

d) Normalisation de l'amplitude

Objectif : Uniformiser le volume des différents fichiers audio afin que le modèle n'interprète pas des écarts de volume comme des caractéristiques du signal vocal.

La normalisation consiste à ajuster l'amplitude des signaux audio pour qu'ils aient tous une même échelle de volume. Certains fichiers peuvent être enregistrés à un volume faible, d'autres à un niveau très élevé, ce qui perturbe les modèles de transcription. Pour y remédier, chaque signal est divisé par sa valeur absolue maximale, ce qui permet de ramener son amplitude dans une plage standardisée entre -1 et 1. Cela garantit que tous les fichiers ont un niveau sonore uniforme, sans saturation ni sous-amplification, ce qui améliore la clarté du signal et la stabilité des performances du modèle sur des données hétérogènes.

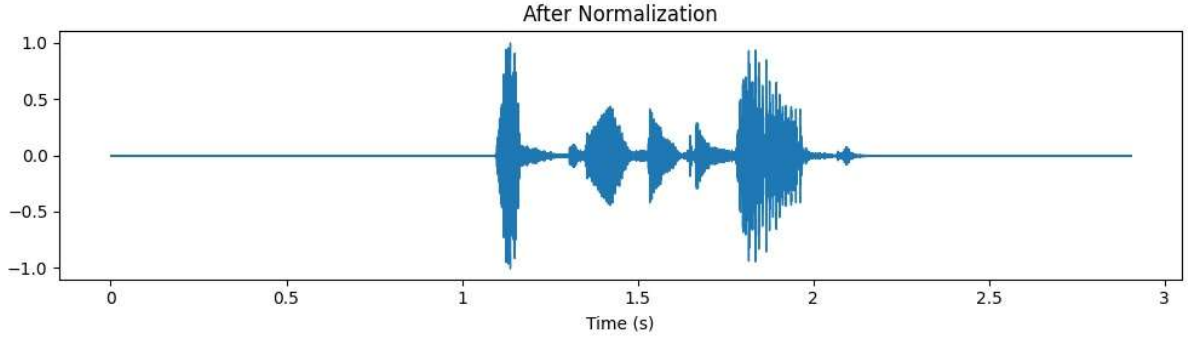

Figure 4: Signal après Normalisation

e) Suppression des silences

Objectif : Supprimer les silences inutiles au début et à la fin du fichier, qui peuvent perturber la détection des segments parlés.

Les silences en début et fin d'enregistrement ne contiennent aucune information utile à la transcription, mais ils peuvent perturber la détection automatique de la parole. Pour les supprimer, une détection automatique de silence a été effectuée à l'aide de la bibliothèque `librosa`. Cette méthode évalue l'énergie du signal sonore dans le temps et coupe les portions où le niveau est inférieur à un seuil défini (par exemple 20 décibels en dessous du maximum). En supprimant ces segments

silencieux, on réduit la taille des fichiers, on accélère le traitement et on recentre l'analyse sur la parole réelle, ce qui améliore la qualité des résultats.

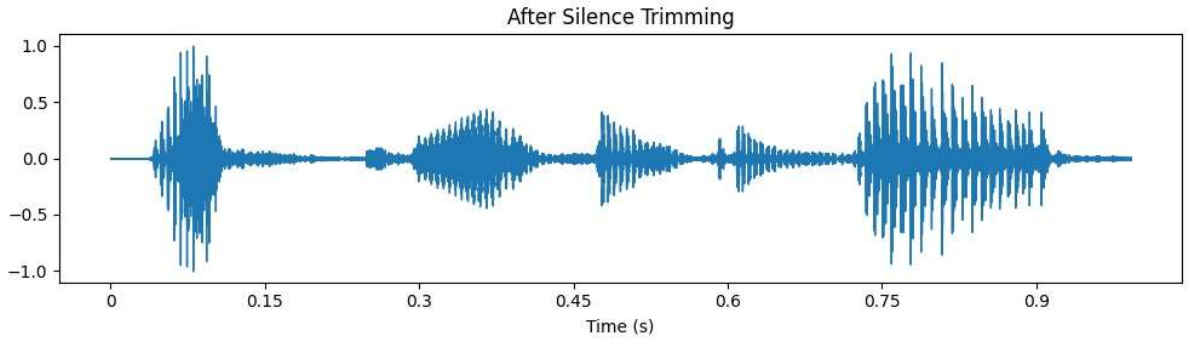

Figure 5: Signal après Suppression des silences

5. Évaluation de la performance de transcription

Une fois les transcriptions générées par le modèle Whisper, il est essentiel d'évaluer leur précision afin de mesurer l'efficacité du système mis en place. Pour cela, nous avons utilisé la métrique Word Error Rate (WER), une mesure standard dans le domaine de la reconnaissance vocale.

Le WER quantifie l'écart entre la transcription produite automatiquement par le modèle et la transcription de référence (texte exact attendu). Il s'exprime sous forme de pourcentage et se calcule comme suit :

$$\mathbf{WER = S+D+I \,/\, N}$$

- S est le nombre de **substitutions** (mots incorrectement transcrits),
- D est le nombre de **suppressions** (mots oubliés),
- I est le nombre d'**insertions** (mots ajoutés à tort),
- N est le nombre total de mots dans la référence.

Dans notre projet, les transcriptions automatiques ont été comparées ligne par ligne avec les textes fournis dans le dataset cv-other-test, et le score WER a été calculé pour chaque enregistrement à l'aide de la bibliothèque **jiwer**. Ces scores ont été ajoutés au tableau de données final pour une analyse plus approfondie.

En complément, une **moyenne globale du WER** a été calculée pour obtenir une mesure d'ensemble de la performance du modèle sur tout le corpus. Cette moyenne constitue un indicateur central permettant d'évaluer l'adéquation du modèle Whisper à notre tâche spécifique et à la qualité des données traitées.

Dans notre cas, la moyenne globale du WER obtenue est de **0,17**, ce qui traduit un taux d'erreur relativement faible et une transcription de qualité acceptable dans l'ensemble. L'évaluation des erreurs permet également d'identifier les cas problématiques : accents particuliers, mauvaise qualité audio, ou mots spécifiques mal reconnus. Ces observations alimentent ensuite la réflexion sur les perspectives d'amélioration.

6. Conclusion

Ce chapitre a permis de détailler les différentes étapes de notre méthodologie, de la préparation des données à l'évaluation des performances du modèle. En nous appuyant sur l'architecture Whisper, nous avons défini une stratégie d'entraînement et d'évaluation adaptée à notre tâche spécifique de transcription automatique. Les choix méthodologiques, notamment le calcul du WER comme indicateur principal, nous ont permis de mesurer de manière rigoureuse la qualité des transcriptions. L'identification des erreurs et des cas problématiques, tels que les accents, la qualité audio ou certains mots spécifiques, constitue un point clé pour orienter les pistes d'amélioration future. La méthodologie mise en place pose ainsi les fondations nécessaires pour l'analyse des résultats, leur interprétation, et l'exploration des solutions envisagées pour perfectionner notre système de transcription automatique.

Chapitre 4 : Implémentation du projet

1. Introduction

Ce chapitre décrit l'ensemble du processus d'implémentation technique du modèle, depuis la préparation de l'environnement de développement jusqu'à l'entraînement et l'évaluation finale. Notre objectif est de transformer la méthodologie définie précédemment en une solution opérationnelle, en exploitant des bibliothèques modernes et des outils performants pour garantir un déploiement efficace.

2. Langages et bibliothèques utilisés

Pour implémenter notre projet, nous avons utilisé :

Langages :

- Python 3.9 : utilisé comme langage principal pour la manipulation des données, la modélisation et le développement de l'application.

Bibliothèques :

- **Pandas** : est une bibliothèque essentielle en science des données pour la manipulation et l'analyse de structures de données tabulaires. Dans ce projet, elle a été utilisée pour charger les fichiers CSV contenant les métadonnées audio (chemin, texte de référence, nom de fichier), mais aussi pour enregistrer les résultats de transcription générés par le modèle.

- **Whisper** : st la bibliothèque officielle proposée par OpenAI pour exploiter leurs modèles de transcription automatique. Elle permet de charger différents modèles (tiny, base, small, medium, large) et de les utiliser via des appels simples pour transcrire des fichiers audio. Dans notre projet, nous avons utilisé le modèle Whisper Medium afin de transformer les fichiers vocaux en texte. La méthode transcribe() fournie par la bibliothèque est au cœur du processus de transcription.

- **Jiwer** : est une bibliothèque spécialisée dans le calcul des erreurs de transcription, notamment le **Word Error Rate (WER)**. Elle compare deux textes (le texte de référence et le texte transcrit) et mesure les différences en termes d'insertion, suppression et substitution de mots.

- **Librosa** : est une bibliothèque puissante pour le traitement de l'audio en Python, largement utilisée dans les projets de reconnaissance vocale et de musique. Dans notre cas, elle a permis de charger les fichiers audio, de les rééchantillonner à 16 kHz (fréquence exigée par Whisper), de normaliser leur amplitude, et de supprimer les silences au début et à la fin.

- **Pydub** : est une bibliothèque de traitement audio de haut niveau qui facilite la conversion de formats (comme MP3, WAV, etc.) et le découpage de fichiers audio. Elle est notamment utilisée lorsque les fichiers audio initiaux ne sont pas au format WAV ou présentent une structure incompatible avec d'autres bibliothèques comme librosa.

- **Scipy**, via son module scipy.signal, fournit des outils pour appliquer des filtres numériques. Nous avons utilisé cette bibliothèque pour effectuer un **filtrage passe-haut** sur les fichiers audio

- **Soundfile** (ou pysoundfile) est une bibliothèque utilisée pour lire et écrire des fichiers audio. Elle prend en charge différents formats comme WAV, FLAC ou OGG. Après les traitements effectués par librosa et scipy, les signaux nettoyés ont été enregistrés en fichiers WAV grâce à soundfile, afin d'être transmis au modèle Whisper dans de bonnes conditions.

3. Présentation des résultats

L'évaluation des performances de notre modèle Whisper a permis d'obtenir des résultats significatifs sur le corpus étudié. Le principal indicateur utilisé, le **Word Error Rate (WER)**, a permis de quantifier l'exactitude des transcriptions générées. La moyenne globale du WER calculée sur l'ensemble des données est de **0,17**, ce qui traduit une qualité de transcription globalement satisfaisante pour notre tâche spécifique.

Les performances du modèle peuvent être détaillées comme suit :

Critère	Résultat
Moyenne globale du WER	0,17
Nombre total de fichiers testés	2962
Taux de reconnaissance correct	82%

Figure 6: Table de résultat

4. Analyse des erreurs

Malgré des performances satisfaisantes, certaines erreurs récurrentes ont été observées :

- **Accents et dialectes** : Le modèle a parfois des difficultés à traiter des accents régionaux ou des variantes de prononciation spécifiques.
- **Mots rares ou techniques** : Les termes spécifiques, notamment les noms propres, sigles, ou termes techniques, sont plus sujets à des erreurs de transcription.
- **Confusions fréquentes** : Certaines erreurs sont récurrentes, comme la confusion entre des mots homophones ou proches phonétiquement.

5. Interface

L'interface développée dans le cadre de ce projet vise à offrir une plateforme simple, intuitive et accessible permettant aux utilisateurs de tester la transcription automatique à partir de leurs propres fichiers audio. Accessible en ligne à l'adresse https://issalmou.github.io/Speechly/, elle constitue la vitrine applicative du pipeline mis en œuvre.

Fonctionnalités principales :

✓ Enregistrement audio : utilisateur peut enregistrer sa propre voie à travers le micro de son ordinateur, cette fonctionnalité est utile pour tester la transcription en temps réel

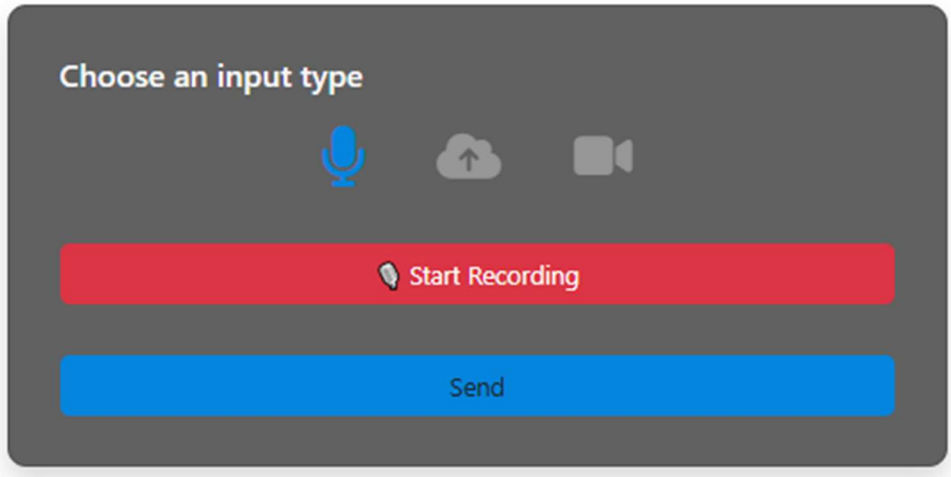

Figure 7: Fonction d'enregistrer un audio

✓ Téléversement d'un audio : utilisateur peut téléverser un audio existant depuis son ordinateur, il peut réécouter l'audio puis l'audio

soumis une transcription. Cette fonctionnalité permet de traiter des fichiers préenregistrer comme les interviews et les conférences.

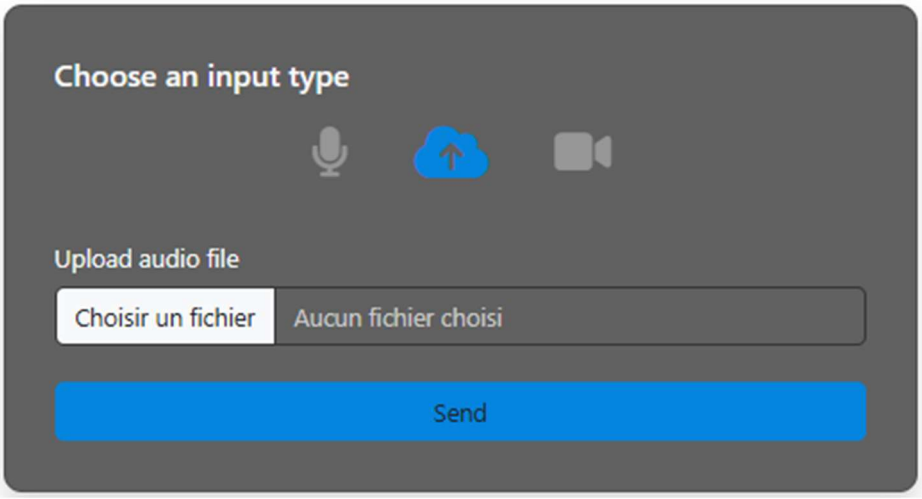

Figure 8:Fonction de téléverser un audio

- ✓ Téléversement d'un fichier vidéo : L'interface accepte également les fichiers vidéo, à partir desquels l'audio est automatiquement extrait. Cette fonctionnalité est utile dans le contexte de la transcription de présentations ou de vidéos éducatives. Le système se charge d'isoler la piste audio pour la transmettre au modèle Whisper. Cela rend l'outil polyvalent, puisqu'il ne se limite pas aux seuls fichiers audio.

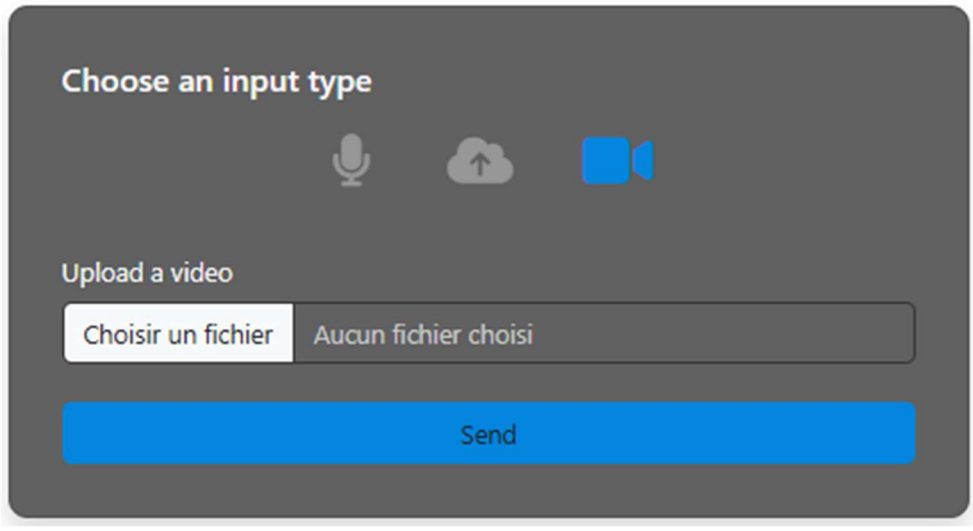

Figure 9:Fonction de téléverser un vidéo

6. Conclusion

L'implémentation du projet a permis de traduire la démarche méthodologique en une réalisation concrète, articulée autour d'un pipeline complet de transcription automatique. De l'importation des données à leur prétraitement audio, en passant par l'application du modèle Whisper et l'évaluation des performances via la métrique WER, chaque étape a été soigneusement mise en œuvre et testée. L'intégration d'une interface utilisateur simple et fonctionnelle a renforcé l'accessibilité de notre système, en offrant aux utilisateurs divers moyens d'interagir avec la transcription (enregistrement, téléversement audio ou vidéo). Cette phase de mise en œuvre constitue ainsi une base solide pour des améliorations futures, qu'il s'agisse d'optimiser les performances, d'adapter l'outil à d'autres langues ou d'enrichir l'expérience utilisateur par de nouvelles fonctionnalités.

Chapitre 5 : Perspectives

Les résultats obtenus, avec un **Word Error Rate (WER) moyen de 0,17**, confirment les performances satisfaisantes du modèle Whisper pour la tâche de transcription sur notre corpus. Cependant, ces résultats mettent en évidence plusieurs axes d'amélioration essentiels pour affiner notre système et le rendre plus robuste.

Les principales perspectives d'amélioration identifiées sont les suivantes :

Enrichissement du corpus :
Il est essentiel de diversifier et d'augmenter la taille du corpus d'entraînement, en incluant davantage de données avec des accents variés, des mots techniques et des contextes spécifiques. Cela permettrait au modèle d'être plus performant face aux particularités linguistiques et aux situations réelles.

Optimisation des paramètres du modèle :
L'ajustement des hyperparamètres (comme le *beam size*, *temperature*, etc.) lors de l'inférence pourrait contribuer à réduire les erreurs et à obtenir des transcriptions plus précises.

Développement d'un système de post-traitement :
La mise en place d'un module de correction automatique, capable de rectifier les erreurs récurrentes (noms propres, homophones, sigles), permettrait d'améliorer la qualité finale des transcriptions.

Exploration de techniques de fine-tuning :
Entraîner ou adapter le modèle sur des corpus spécifiques à notre domaine (par exemple, des enregistrements techniques ou scientifiques) pourrait renforcer la précision sur des termes peu fréquents ou spécialisés.

En résumé, bien que le WER moyen de **0,17** reflète une performance encourageante, ces perspectives montrent que des efforts supplémentaires sont nécessaires pour améliorer la robustesse et l'adaptation du système à des contextes plus complexes et variés. Ces pistes ouvrent la voie à un perfectionnement continu de notre solution de transcription automatique.

Conclusion générale

Ce rapport a permis d'explorer en profondeur les aspects techniques et théoriques liés au développement d'une application de transcription automatique de l'audio vers le texte, basée sur le modèle Whisper. L'objectif principal était de démontrer notre capacité à mettre en œuvre une solution complète, tout en valorisant nos compétences techniques, notre sens de l'analyse, et notre aptitude à travailler en équipe.

Tout au long de ce projet, nous avons enrichi notre savoir en manipulant des outils avancés, en relevant des défis concrets, et en prenant des décisions critiques à chaque étape du processus. Malgré ces limitations, nous avons réussi à contourner les obstacles et à mener à bien le développement de notre solution.

Cette expérience s'est révélée extrêmement formatrice. Elle nous a permis non seulement de consolider nos acquis, mais aussi de développer un regard critique sur les modèles de traitement automatique de la parole. Désormais, nous envisageons avec enthousiasme la poursuite de ce travail, notamment à travers l'optimisation du modèle, l'enrichissement de l'interface utilisateur et l'intégration de nouvelles fonctionnalités.

Nous restons convaincus que ce projet constitue une base solide pour des applications réelles dans divers domaines, et nous sommes motivés à continuer à progresser dans le domaine de l'intelligence artificielle, en participant activement à son évolution.

Bibliographie

- OpenAI. (2022). *Whisper: Speech recognition model*. GitHub Repository. https://github.com/openai/whisper
- Jiwer Documentation. (n.d.). *Jiwer: Word Error Rate computation*. https://github.com/jitsi/jiwer
- Google Colab. (n.d.). *An interactive Python notebook environment*. https://colab.research.google.com
- McFee, B. et al. (2015). *librosa: Audio and Music Signal Analysis in Python*. Proceedings of the 14th Python in Science Conference, 18–24.
- ffmpeg-python. (n.d.). *FFmpeg Python Bindings*. https://github.com/kkroening/ffmpeg-python
- Pydub. (n.d.). *Manipulate audio with a simple and easy high-level interface*. https://github.com/jiaaro/pydub
- SciPy Community. (2022). *SciPy Library for Scientific Computing*. https://scipy.org/
- Hugging Face. (n.d.). *Faster-Whisper: Optimized inference for Whisper models*. https://github.com/guillaumekln/faster-whisper
- Common Voice : https://commonvoice.mozilla.org/en/datasets