UNIVERSITÉ
LAVAL

Proposition de projet

Réalisé par
Elise Beaussart (537 186 763)
Hugo Breniaux (537 188 162)

Dans le cadre du cours
IFT-7030 (NRC : 85789)

Travail présenté à
Mr Cem Subakan

Département d'informatique et de génie logiciel
Faculté des Sciences et de Génie
Université Laval
Date de remise : 18 octobre 2023

Résumé

Notre projet repose sur la conviction que l'intelligence artificielle (IA) peut jouer un rôle majeur dans l'amélioration de l'apprentissage des langues étrangères. Nous visons à créer un outil performant pour aider les étudiants et les étudiantes à pratiquer une langue étrangère, en mettant l'accent sur l'acquisition la correction d'erreurs grammaticales.

La clé de l'apprentissage efficace d'une langue étrangère réside dans la pratique orale. Nous envisageons donc le développement de modèles d'IA capables de prendre en entrée des enregistrements audio d'utilisateurs et de corriger les éventuelles erreurs grammaticales présentes. Ces modèles restitueront ensuite l'audio d'origine, mais avec les corrections nécessaires.

Pour concrétiser notre projet, nous allons nous appuyer sur des modèles pré-existants, que nous améliorerons et affinerons en fonction de nos besoins spécifiques. Notre objectif est de spécialiser ces modèles pour qu'ils comprennent un accent particulier. Pour ce faire, nous utiliserons des données audio provenant de locuteurs parlant anglais avec un accent distinct (lié à leurs origines). Cela permettra à nos modèles de se spécialiser dans la compréhension des utilisateurs d'une origine spécifique s'exprimant en anglais, permettant de contourner l'une des limitation actuelle de la reconnaissance automatique de la parole ASR qui est liée aux accents.

Notre architecture globale consistera en une séquence de modules : données audio en entrée, conversion de la parole en texte, correction des erreurs grammaticales, puis synthèse vocale. Nous visons à obtenir des taux de réussite élevés, grâce au fine-tuning qui nous permettra d'améliorer considérablement les performances des modèles pré-entraînés généraux. Finalement, nous obtiendrons une chaîne de traitement où la sortie d'un modèle devient l'entrée de l'autre.

Table des matières

1 Introduction

L'apprentissage d'une langue est un long et fastidieux processus. Il est tout de même possible grâce à l'enseignement, de rendre cette tâche plus aisée. Celui-ci se caractérise notamment par le biais de professeurs, mais aussi via l'utilisation d'outils (numériques ou non) pour permettre de guider les élèves et de leur faciliter l'apprentissage.

Ce projet a pour objectif d'améliorer le processus d'apprentissage de l'anglais grâce à des outils numériques de traitement du signal et de traitement du langage naturel (NLP) en corrigeant la prononciation et la grammaire de l'élève à travers la parole. La population premièrement ciblée sont les personnes ayant comme langue maternelle le français et souhaitant apprendre l'anglais.
Pour mener à bien ce projet, on peut mettre en place un pipeline qui consiste à :

1. Transcrire fidèlement les paroles de l'élève en texte à l'aide d'un modèle Speech-To-Text (STT) tel que Speechbrain speech-to-text, Google cloud speech-to-text, ...
2. Détection et correction des erreurs grammaticales du texte à l'aide d'un modèle Large Language Model (LLM) tel que BERT, GPT, ...
3. Synthèse de la parole en convertissant le texte corrigé en audio à l'aide d'un modèle Text-To-Speech (TTS) tel que Speechbrain, coqui-ai, mozilla-tts, ... Cela permettra à l'élève d'entendre la prononciation du texte corrigé.

Lors de ces différentes tâches, nous allons nous baser sur des modèles déjà existants (cités précédemment). On va tenter de les améliorer en les affinant en fonction de nos besoins :

— Sur le STT, l'affinage se concentrera sur les accents (anglais avec accent français), ainsi que sur la précision et la fidélité de la transcription pour bien récupérer les possible erreurs grammaticales.
— Sur le LLM, l'affinage sera orienté sur la détection des erreurs grammaticales et la correction pour avoir en sortie un modèle Grammatical Error Correction (GEC).

Ce document va être composé comme suit, on va en premier lieu réaliser une revue de la littérature pour situer et comparer notre solution avec les techniques existantes. On va par la suite étudier les différents jeux de données qu'il nous serait possible d'utiliser, ainsi que les mesures de performances adaptées et les ressources informatiques nécessaires.

2 Revue de la littérature

Notre POC contiendra donc les tâches suivantes : STT, GEC, TTS. Ce sont des tâches déjà utilisées dans divers contextes (assistants vocaux et virtuels, domaine de la santé, outils d'accessibilité, etc.). L'éducation émerge comme un domaine d'application en croissance, car la GEC permet de personnaliser l'apprentissage et de s'auto-corriger pour s'améliorer. Toutefois, les recherches existantes sur l'adaptation du STT et TTS sont éparpillées dans divers domaines d'application sans se concentrer particulièrement sur la pédagogie. [4] Nous allons donc organiser notre revue de littérature d'abord pour le domaine du STT et TTS, et ensuite pour le GEC.

La conversion de la parole en texte et la reconnaissance automatique de la parole ont connu des avancées significatives ces dernières années, principalement grâce aux progrès fulgurants dans les domaines de l'apprentissage en profondeur et des approches basées sur les données. Le STT repose sur trois piliers majeurs : la modélisation acoustique, la modélisation linguistique et le décodage. La modélisation acoustique a connu une évolution significative, passant des Modèles de Markov Cachés (HMM) traditionnels à l'utilisation de techniques d'apprentissage en profondeur telles que les Réseaux de Neurones Convolutifs (CNN), les Réseaux de Neurones Récursifs (RNN) et les modèles basés sur les Transformers. Ces avancées se sont traduites par des améliorations notables en termes de précision, notamment dans des environnements bruyants et pour des accents variés.

La modélisation linguistique, quant à elle, se concentre sur la compréhension du contexte linguistique des signaux vocaux. Si les modèles n-grammes traditionnels prédominaient, des avancées récentes ont exploré des approches plus avancées, comme les RNN, les réseaux LSTM (Long Short-Term Memory) et les modèles basés sur les Transformers. Ces avancées ont considérablement renforcé la capacité de gestion de la parole conversationnelle, des termes hors vocabulaire et du langage spécifique à des domaines particuliers.

Le décodage, troisième composante essentielle, consiste à traduire la sortie des modèles acoustiques et linguistiques en texte final. Les méthodes traditionnelles, comme le décodage de Viterbi basé sur les Modèles de Markov Cachés, ont été remplacées par les systèmes ASR de bout en bout, qui utilisent des techniques d'apprentissage en profondeur pour mapper directement la parole en texte. Ces systèmes ont démontré des résultats prometteurs en termes de précision et d'efficacité.[3]

En ce qui concerne la synthèse vocale, on retourve aussi différentes approches. La première approche est le TTS concaténatif qui repose sur l'enregistrement préalable de la parole humaine, suivi de la concaténation de petites unités de parole pour générer une parole synthétisée. Cette méthode est réputée pour sa capacité à produire une parole de haute qualité et naturelle. Cependant, elle présente des inconvénients majeurs, notamment le besoin de disposer d'une grande quantité de données de parole enregistrée, ainsi que des limitations dans la génération de variations de ton et de hauteur.

Une autre approche traditionnelle, la synthèse de formants, consiste à modéliser le tractus vocal pour générer la parole en manipulant les fréquences des formants des sons de parole. Cette méthode offre un meilleur contrôle sur des aspects tels que la hauteur et le timbre. Toutefois, elle peut parfois manquer de naturel et de réalisme.

Le TTS par sélection d'unités est une approche hybride qui combine des techniques de concaténation et de synthèse de formants. Cette méthode implique la sélection et la concaténation de petites unités de parole préenregistrée, tout en appliquant des techniques de synthèse de formants pour ajuster la parole. Cette approche permet de générer une parole de haute qualité et naturelle.

Mais récemment, les travaux de recherche se concentrent sur le TTS basé sur l'apprentissage en profondeur. Elle fait appel, encore une fois, à des réseaux neuronaux tels que les RNNs, les CNNs et les modèles basés sur les transformers pour générer la parole. Ces modèles ont la capacité d'apprendre à partir de grandes quantités de données, leur permettant de produire une parole d'une grande précision, avec une sonorité naturelle et une grande expressivité.

Enfin, l'augmentation du nombre d'apprenants de la langue anglaise met en évidence la nécessité de développer des ressources automatisées ayant un impact pédagogique plus étendu. Parmi ces ressources, les efforts de recherche dans le domaine de la correction automatique des erreurs grammaticales (GEC) occupent une place importante. Jusqu'à présent, les approches basées sur des règles et les modèles de traduction automatique, tels que les modèles de traduction statistique basée sur des phrases, ont été privilégiés dans ce domaine. Cependant, il existe un intérêt marqué pour le développement de modèles de réseaux neuronaux capables de surmonter les limitations de ces approches et d'améliorer les résultats.[5]

En somme, notre projet s'inscrit pleinement dans l'ère actuelle car nous avons choisi de mettre en œuvre une approche neuronale pour aborder nos trois principales tâches. Notre approche consiste à exploiter des modèles pré-entraînés, une pratique largement adoptée aujourd'hui, que nous allons ensuite affiner pour répondre à nos besoins spécifiques. Nous souhaitons traiter de manière spécifique l'un des défis des systèmes de STT : la reconnaissance des accents.

3 Jeux de données

Comme indiqué dans l'introduction, nous allons mettre en place un pipeline entre nos différents modèles. Voici à quoi celui-ci ressemble :

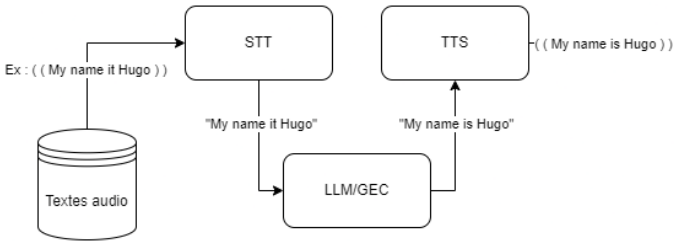

FIGURE 1 – Pipeline

En entrée, nous aurons des données audio (des paroles avec des erreurs grammaticales ou non), et en sortie, nous aurons des paroles corrigées si nécessaire.

On va utiliser plusieurs datasets pour faire fonctionner ce pipeline :

- **Dataset auto-généré** : Comme il y a très peu de datasets contenant des paroles en anglais avec des erreurs grammaticales, il a été décidé de créer un petit échantillon (environ une centaine) de paroles nous-même. Comme ça, nous aurons des phrases en anglais avec un accent français ; des transcriptions "fausses" (avec des erreurs) et leurs corrections ; des transcriptions "justes" (sans erreur). En faisant cela, on peut tenter d'affiner les modèles sur ces données, ainsi que tester les modèles.
- **JFLEG** : Jeu de données contenant des phrases erronées et leurs corrections (Corpus GEC). Composé de plus de 1500 items (754 pour l'entrainement et 747 pour test).
- **Librispeech** : Jeu de données audio contenant plus de 1000h d'audiobook en anglais avec la transcription.

Il est aussi possible de générer synthétiquement nos données pour le GEC comme par exemple avec l'utilisation de C4_200M, ou encore pour nos données audio (Faire un TTS sur des données erronées pour les passer dans notre pipeline).

Toutes ces données nous sont utiles pour l'affinage des modèles pré-entrainés, mais aussi pour les tester (Voir la partie Mersures de performances).

4 Mesures de performances

On va performer différentes évaluations sur notre pipeline. Premièrement, on va lors de l'affinage, se concentrer sur l'évaluation de chacun des modèles individuellement.

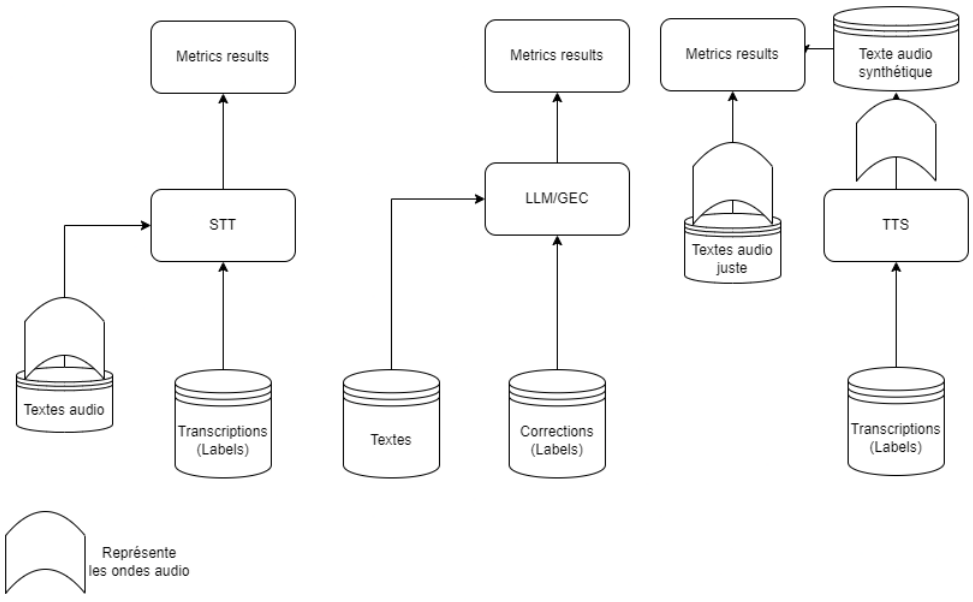

FIGURE 2 – Test des modèles

- STT sera évalué en utilisant Word Error Rate [1], car il évalue le pourcentage d'erreurs mots pour mots de la transcription. On peut aussi utiliser Bilingual Evaluation Understudy pour évaluer la qualité de la transcription.

— GEC sera quant à lui évalué en utilisant la Global Language Evaluation, car il est très utilisé pour l'évaluation de la qualité de correction en termes de cohérence et la fluidité. On peut aussi utiliser le F1-score, qui permet de vérifier que des erreurs n'ont pas été insérées.
— TTS est difficilement évaluable. On peut tout de même vérifier qu'un signal d'entrée soit bien répliqué en utilisant sa version originale et synthétisée, notamment en calculant le Mel Cepstral Distortion (MEL). Il est aussi possible d'évaluer "manuellement" en utilisant Mean Opinion Scores [2]. C'est un score entre 1 et 5, évalué par un ou plusieurs êtres humains.

5 Ressources informatique

En terme de ressources informatiques nous disposons de Google Colab, ainsi que d'un serveur Jupyter. Cela est supposé suffisant car nous n'allons pas entrainer de zéro des modèles, mais utiliser des modèles déjà existants.
Grâce à Google Colab nous allons pouvoir suivre les ressources utilisée (RAM, et disque). Nous pouvons aussi implémenter différentes méthodes pour évaluer le temps d'exécution de chacuns des modèles.

6 Liste de contrôle

— **Pour être réaliste** : Nous disposons déjà de modèles entrainés, donc les ressources demandées seront moindres. Aussi, nous ne travaillerons pas avec des centaines de giga de données, déjà car il n'y en a pas autant de disponible, et car ce n'est pas nécessaire d'en avoir autant (modèles pré-entraînés). Dans ce projet nous allons réaliser un POC, pour principalement tester nos modèles et les données utilisées et les opérations les plus coûteuses seront l'affinage, et la correction de grammaire.
— **Pour avoir un ensemble de données bien défini :** Nous disposons de modèles pré-entraînés, comme cité ci-dessus. Ce qui fait que nous n'avons pas besoin d'énormément de données. De plus, nos données seront labellisées et cohérentes avec notre utilisation, surtout si c'est nous qui en générons.
— **Pour définir clairement quelle sera l'entrée et quelle sera la sortie :** Cela dépend de ce que l'on souhaite faire. On va utiliser des données labellisée pour tester nos modèles, néanmoins en situation réelle nous n'en avons pas. Mais ce qu'on veut en entrée (son) est identique à ce qu'on veut en sortie (son corrigé ou non).
— **Pour ne pas être trivial, ni même être quelque peu ambitieux ou intéressant :** Dans notre cas, bien que les modèles existent déjà pour différents cas d'utilisation, nous n'avons pas trouvé durant nos recherches des projets similaires à ce que nous allons réaliser. C'est aussi dû au fait qu'il n'y ai pas beaucoup de jeu de données disponibles, augmentant la complexité de la réalisation de ce projet. Il est aussi possible de l'appliquer dans plusieurs domaines, comme dans l'accessibilité pour les personnes handicapées, dans l'éducation...
— **Pour être dans le domaine de l'apprentissage automatique et/ou du traitement du signal :** Le STT et TTS sont deux parties appartenant pleinement au traitement du signal. Bien que GEC fasse partie de NLP, il peut aussi être inclus dans le domaine du traitement du signal. Même si nous ne construisons pas les modèles directement, nous les affinerons, ce qui là aussi reste dans la thématique du traitement du signal.
— **Pour être bien situé dans la littérature existante :** Comme cité précédemment, durant nos recherches aucun autre projet similaire au nôtre n'a été trouvé. Néanmoins, nous avons concentré nos recherches sur les modèles composant notre pipeline, qui eux existent déjà. De ce fait nous avons dû comparer les différents modèles existants pour qu'ils puissent d'intégrer au pipeline.

Glossaire

ASR acronyme anglais de Automatic Speech Recognition (fr : Reconnaissance automatique du langage). 1, 4

CNN acronyme anglais de Convolutional Neural Network (fr : Réseau de neurones convolutif). 3, 4

GEC acronyme anglais de Grammatical Correction Error (fr : Correction d'erreur grammaticale). 3–6

HMM acronyme anglais de Hidden Markov Model (fr : Modèle de Markov caché). 3

LLM acronyme anglais de Large Language Model (fr : Large modèles de langage). 3

NLP acronyme anglais de Natural Language Processing (fr : Traitement du langage naturel). 3, 6

POC acronyme anglais de Proof Of Concept (fr : Preuve de concept). 3, 6

RNN acronyme anglais de Recurrent Neural Network (fr : Réseau de neurones récurrents). 3, 4

STT acronyme anglais de Speech-To-Text (fr : Parole à texte). 3–6

TTS acronyme anglais de Text-To-Speech (fr : Texte à parole). 3–6

Références

[1] Ryan Connor. How to evaluate speech recognition Models. *News, Tutorials, AI Research*, 9 2023.

[2] Hugging Face. Evaluating text-to-speech models - Hugging face audio course.

[3] V. Madhusudhana Reddy, T. Vaishnavi, and K. Pavan Kumar. Speech-to-text and text-to-speech recognition using deep learning. In *2023 2nd International Conference on Edge Computing and Applications (ICECAA)*, pages 657–666, 2023.

[4] Daniel Markgraf Susanne Robra-Bissantz Ricarda Schlimbach, Heidi Rinn. A litterature review on pedagogical conversationnal agent adaptation. 2022.

[5] Ayush Gupta Yokila Arora, Jaspreet Kaur. Grammaticalerrorcorrectionusingneuralnetworks.