Conception d'un système de reconnaissance de la parole pour le théâtre

Emma Martinez

Conception d'un système de reconnaissance de la parole pour le théâtre

Emma

MARTINEZ

Sous la direction de Benjamin LECOUTEUX et Rémi RONFARD

Laboratoire : LIG

UFR LLASIC

Département Sciences du Langage & FLE

Section IDL

Mémoire de master Sciences du Langage

Parcours Industrie de la langue

Année universitaire 2022-2023

Conception d'un système de reconnaissance de la parole pour le théâtre

Emma

MARTINEZ

Sous la direction de Benjamin LECOUTEUX et Rémi RONFARD

Laboratoire : LIG

UFR LLASIC

Département Sciences du Langage & FLE

Section IDL

Mémoire de master Sciences du Langage

Parcours Industrie de la langue

Année universitaire 2022-2023

Remerciements

Je tiens à exprimer ma sincère gratitude envers Solange Rossato, mon enseignante référente pour ses conseils éclairés, son soutien continu et sa guidance tout au long de mon stage.

Je remercie également Benjamin Lecoûteux et Rémi Ronfard, mes co-encadrants pour leur présence durant ce stage. Leurs conseils, leur expertise et leur patience ont été d'une grande aide.

Je remercie aussi Solène Evain et Eric Leferrand qui, aux côtés de Mme Rossato, ont été des appuis précieux.

Enfin, mes remerciements à ma mère pour sa relecture attentive de mon travail et pour son soutien.

DÉCLARATION ANTI-PLAGIAT

1. Ce travail est le fruit d'un travail personnel et constitue un document original.
2. Je sais que prétendre être l'auteur d'un travail écrit par une autre personne est une pratique sévèrement sanctionnée par la loi.
3. Personne d'autre que moi n'a le droit de faire valoir ce travail, en totalité ou en partie, comme le sien.
4. Les propos repris mot à mot à d'autres auteurs figurent entre guillemets (citations).
5. Les écrits sur lesquels je m'appuie dans ce mémoire sont systématiquement référencés selon un système de renvoi bibliographique clair et précis.

PRENOM : Emma

NOM : Martinez

DATE :23/08/2023

Sommaire

Introduction

L'avènement des technologies de reconnaissance automatique de la parole (ASR, venant du terme “Automatic Speech Regnition” en anglais) a révolutionné de nombreux domaines de la vie courante, allant des applications mobiles jusqu’aux outils de domotiques en passant par les moyens de télécommunications. Parallèlement, le théâtre, en tant qu’espace majeur de l'expression culturelle, a toujours été un espace d'innovation, mêlant tradition et modernité. Pourtant, en dépit des avancées de l'ASR, le monde théâtral, en particulier dans le contexte français, est resté en grande partie inexploré en ce qui concerne l'application de ces technologies. Ce manquement actuel représente une opportunité, car le théâtre pose des défis uniques en termes d'intonations, de dialogues et d'émotions. L'objectif de ce stage est de mettre en œuvre un système ASR adapté aux nuances du théâtre français. Pour ce faire, j’adopterai une approche méthodologique combinant l'évaluation de la technologie existante et le développement d'algorithmes et de techniques l'utilisant d’une façon optimale. En se plaçant en précurseurs de ce domaine naissant, nous espérons créer un nouvel outil se basant sur l'union entre technologie et art.

Le sujet en lui-même aborde la question de l’ASR au théâtre par la conception d’un système d’alignement entre le script de théâtre et l’enregistrement de la pièce de théâtre elle-même, autrement dit ce qui a été prononcé réellement. En d’autres termes, il a notamment pour but de créer un outil permettant de sous-titrer automatiquement les enregistrements des pièces de théâtre. Ledit outil sera dans l’idéal assez robuste : il pourra être appliqué à d’autres corpus de pièces de théâtre que celles sur lesquelles nous souhaitons travailler, et pourra être appliqué à d’autres langues telles que l’anglais.

Un réel besoin de la part des pratiquants de théâtre, aussi bien pour les metteurs en scène, que pour les acteurs et les spectateurs existe. D’une part, dans une optique d’observation analytique pour le metteur en scène et les comédiens. Autrement dit, ces derniers pourront observer lorsque la pièce sous-titrée sera diffusée, la distance entre ce qui est réellement dit et le script de théâtre. D’autre part, le public concerné par la visualisation de cette pièce pourra également trouver une utilité dans le sous-titrage. En effet, pour un public malentendant ou pour des paroles prononcées difficiles à percevoir, les spectateurs pourront s’appuyer sur le sous-titrage pour avoir une meilleure compréhension. Enfin, si le sous-titrage n’est pas

optimal, il pourra tout de même être une base de travail considérable pour un annotateur professionnel, il pourra ainsi gagner du temps en ayant pour seule tâche de réajuster le travail du système.

L'approche que je vais utiliser se base sur des outils préexistants dans le domaine de l'informatique et plus précisément du TAL. La méthode de création du système consiste à utiliser un outil de décodage de la parole disponible au sein de l'équipe ou en libre accès sur Internet. On peut citer Whisper ou Speechbrain qui sont tous deux installables dans Python et disponibles en ligne. Une fois le décodage effectué, il faut ensuite aligner son résultat avec le script réel de la pièce de théâtre.

Dans un premier temps, je dresserai le cadre de cet ouvrage en présentant l'environnement d'accueil et l'organisation du stage. Cela permettra d'ancrer le travail dans un contexte précis et de mieux comprendre les circonstances de cette recherche. Ensuite, je réaliserai un bref état de l'art sur la parole et la reconnaissance automatique de la parole (ASR) dans le contexte théâtral et aussi dans des domaines connexes. Cette section fournira les bases théoriques nécessaires et mettra en lumière les recherches antérieures, soulignant l'importance et la nouveauté de ce travail. Par la suite, la méthodologie employée pour ce mémoire sera exposée en détail. Elle comprendra les outils, techniques et approches utilisés pour concevoir et mettre en œuvre le système ASR pour le théâtre. Enfin, je me concentrerai sur l'évaluation du système proposé, en présentant les résultats obtenus, leur signification et leur impact sur le champ de l'ASR théâtral.

Partie 1

-

Contexte et objectifs du stage

1. La structure d'accueil

Le stage se déroule du 8 mars au 31 juillet dans le bâtiment IMAG se situant au 700 Av. Centrale, 38400 Saint-Martin-d'Hères sur le campus.

Figure 1. Photographie du bâtiment IMAG provenant du site de l'IMAG (n. d.). Récupéré de https://batiment.imag.fr/

Ce bâtiment se dresse comme figure emblématique de la recherche informatique à Grenoble. Il se distingue par ses cinq laboratoires comportant eux-mêmes différentes équipes de travail incluant doctorants, chercheurs, ingénieurs et stagiaires. Il forme ainsi un épicentre de recherche interdisciplinaire. Parmi ces laboratoires, on compte l'Agence pour les Mathématiques en Interaction avec les Entreprises et la Société (AMIES), qui met l'accent sur l'interaction entre les milieux académique et industriel. Il y a également GRICAD, abréviation de Grenoble Alpes Recherche Infrastructure de CAlcul Intensif et de Données, spécialisé dans le traitement de données à grande échelle. Le Laboratoire d'Informatique de Grenoble (LIG), y apporte son expertise en divers domaines de l'informatique, tandis que le Laboratoire Jean

Kuntzmann (LJK) se concentre sur les mathématiques appliquées et informatiques. Enfin, VERIMAG, connu pour ses travaux sur les systèmes embarqués, complète ce tableau.

En regroupant ces entités sous une seule bannière, l'IMAG renforce sa position non seulement comme un pilier de la recherche à Grenoble, mais aussi comme un acteur influent sur la scène internationale de la recherche en informatique et mathématiques appliquées.

2. L'équipe de travail

Le Groupe d'Étude en Traduction Automatique/Traitement Automatisé des Langues et de la Parole (GETALP), où s'est déroulé mon stage, est sous la direction de François Portet. Né de la fusion en 2007 des équipes GETA et GEOD, GETALP est depuis devenu un acteur notable dans le domaine de la communication et du traitement de l'information multilingue, tant à l'écrit qu'à l'oral.

Figure 2. Logo de l'équipe GETALP provenant du site de Getalp (2011). Récupéré de https://lig-getalp.imag.fr/fr/accueil/

Le groupe s'efforce d'aborder une variété d'aspects liés à son domaine, englobant des questions théoriques, méthodologiques et concrètes. L'une des caractéristiques distinctives de GETALP est sa composition multidisciplinaire. En effet, l'équipe rassemble des spécialistes de différents horizons, tels que la linguistique, l'informatique et la phonétique. Cette combinaison d'expertises offre une approche équilibrée et nuancée des défis inhérents à la traduction automatique et au traitement des langues.

3. Le déroulement du stage

Mon stage s'est tenu en salle 329, du lundi au vendredi, de 9h à 17h, avec une heure de pause à midi. Même si nous n'avions pas d'horaires fixes pour nos réunions, des points d'étape réguliers avec mes deux encadrants étaient de mise. Ces rendez-vous nous permettaient de suivre mon avancement, d'évaluer les méthodes utilisées et d'identifier les prochaines étapes.

Chaque semaine, nous organisions également une réunion à part en présence de mon enseignante référente et d'un ou deux doctorants du laboratoire. Ces échanges étaient enrichissants, me permettant de mieux comprendre les nuances des systèmes ASR et de réajuster mes approches si nécessaire.

L'ambiance au laboratoire était collaborative, l'entraide était le maître-mot. Qu'il s'agisse de résoudre un problème technique ou de se familiariser avec un nouvel outil, les doctorants et autres stagiaires étaient toujours prêts à apporter leur aide s'ils en avaient la capacité. Lorsqu'ils n'étaient pas disponibles, ils faisaient toujours en sorte de trouver un créneau pour pouvoir trouver une solution au problème. C'est pourquoi, même en l'absence de mon tuteur ou s'il était indisponible, je savais que je n'étais jamais seule.

L'encadrement du stage a été assuré principalement par Benjamin Lecouteux, professeur-chercheur au sein de l'équipe GETALP. Son co-encadrant se nomme Rémi Ronfard qui est directeur de recherche à l'Inria de Grenoble a également participé au bon déroulement du stage. De plus, l'enseignante référente du stage, Solange Rossato, tout comme Benjamin Lecouteux, est professeur-chercheuse au sein de l'équipe GETALP et participait à l'encadrement.

4. Sujet traité et missions

La personne qui a proposé le stage est l'encadrant du stage Benjamin Lecouteux comme évoqué précédemment. Les sujets qu'il a abordé concernent principalement la parole, plus particulièrement la reconnaissance de la parole ainsi que toutes les thématiques liées aux réseaux de neurones. Il s'agissait à l'origine d'une proposition de la part de Rémi Ronfard, directeur de recherche à l'Inria de Grenoble.

Ce stage intitulé *Reconnaissance de la parole pour le théâtre* a donc été proposé par Rémi Ronfard et Benjamin Lecouteux et provient d'un besoin de la part du monde du théâtre de pouvoir accéder à des alignements entre le script et ce qui a été dit par les comédiens.Cela facilite le sous-titrage afin de permettre à un transcripteur d'accéder à une pièce pré-traitée, qu'il devra seulement modifier, et également de faire la comparaison entre ce qui est dit du script de théâtre et ce qui ne l'est pas.

5. Le thème étudié

La reconnaissance de la parole au théâtre est un domaine peu exploré bien que des thématiques connexes l'aient été. Lorsqu'on souhaite aborder les systèmes ASR pour le théâtre, il s'agirait tout d'abord de définir la notion de parole au théâtre. Ce thème a été très peu étudié, mais on peut simplement partir du principe qu'il s'agit de parole préparée qui tente de reproduire des contextes de parole spontanée, les dialogues. En ce sens, elle se place réellement à mi-chemin entre spontanée et préparée. Cela signifie que sa nature assez spécifique doit être prise en compte pour les systèmes. Il faut d'ailleurs prendre en compte le fait que la parole théâtrale peut contenir des phases dédiées à l'improvisation.

Le seul élément qui s'approche de notre thématique est un article traitant de l'ASR sur du théâtre grec, NLP-Theatre: *Employing Speech Recognition Technologies for Improving Accessibility and Augmenting the Theatrical Experience*, (Katsalis et al., 2022) . Il s'agit d'un système qui automatise les sous-titres en temps réel pour les pièces de théâtre. Le système utilise une méthode pour aligner les sous-titres existants avec une nouvelle piste audio en temps réel, qui peut avoir des dialogues différents. Cette méthode repose sur la reconnaissance vocale en temps réel et suggère des ajustements en fonction de l'estimation du sous-titre actuel et du moment où le sous-titre actuel et le suivant doivent commencer. Bien que ce ne soit pas dans la langue que nous souhaitons traiter et que l'objectif ne soit pas exactement le même, ce papier donne malgré tout une vision possible de ce que peut être notre système.

Partie 2

-

Etat de l’art

Le but premier de ce stage est de développer un outil intégré pour aligner des scripts de théâtre avec leur transcription automatique et ainsi générer des sous-titres de façon automatique. Il s'agit là d'un enjeu majeur pour le domaine du théâtre et de l'ASR. Cet objectif implique l'exploration, le test et l'optimisation de divers outils déjà existants pour aboutir à la solution la plus efficace. L'axe majeur du stage est l'exploitation d'outils ASR open-source ou accessibles au sein de l'équipe. De ce fait, plusieurs interrogations ainsi que plusieurs problématiques se sont présentées lors de leurs utilisations.

1. L'existant au théâtre

Les seules expériences existantes à propos de ce qu'est l'ASR proviennent d'un unique article scientifique cité précédemment, NLP-Theatre: *Employing Speech Recognition Technologies for Improving Accessibility and Augmenting the Theatrical Experience*, (Katsalis et al., 2022). Bien qu'unique, celui-ci propose un objectif similaire au stage. On cherche en effet à aligner des sous-titres avec une piste audio. Le système utilise trois méthodes différentes pour aligner les sous-titres avec une nouvelle piste audio, en se basant sur la similarité de chaînes de caractères, les LSTM pour les caractères et l'extraction de mots-clés. Trois scenarii ont été utilisés pour l'évaluation du système, cela inclut des simulations et des performances réelles de pièces de théâtre. Des évaluations subjectives et objectives ont été réalisées. Une évaluation subjective a été réalisée avec 250 sujets experts , qui ont noté différents aspects de l'expérience. Les résultats ont montré une corrélation significative entre les mesures de similarité objectives grâce à l'IoU et les notes attribuées par les participants. En d'autres termes, lorsque les performances du système étaient mauvaises selon l'évaluation objective, les participants donnaient des notes basses. L'IoU, ou Intersection over Union est une mesure d'alignement assez répandue dans le domaine scientifique notamment pour la vision par ordinateur et détection d'objets. [1]

Pour l'ensemble de données de films grecs, une contribution notable aux erreurs de performances du système concerne les propositions erronées sur le début du prochain sous-titre et le moment où le sous-titre actuel aurait dû commencer. Ces erreurs ont un impact important sur les performances du système pour cet ensemble de données.

[1] Zhou, D., Fang, J., Song, X., Guan, C., Yin, J., Dai, Y., & Yang, R. (2019). IoU Loss for 2D/3D Object Detection. *2019 International Conference on 3D Vision (3DV)*, 85-94. https://doi.org/10.1109/3DV.2019.00019

Les résultats ont montré que le système offrait une expérience synchronisée, mais sa flexibilité était limitée en raison de son incapacité à prendre en compte l'improvisation théâtrale. L'article conclut en soulignant la pertinence de ces systèmes pour améliorer l'accessibilité et l'expérience théâtrale, et propose des pistes pour des travaux futurs, notamment l'intégration de systèmes ASR personnalisés.

Tout d'abord, afin de comprendre quels sont les enjeux des systèmes que nous utilisons, il semble nécessaire de définir ce qu'est réellement la parole au théâtre. Malgré les nombreuses recherches dans différentes langues. Il ne semble pas exister d'ontologie fiable et fixée de la parole au théâtre : cette thématique n'a pas été beaucoup étudiée ce qui implique qu'il n'y a pas réellement de données sur le sujet. Cependant, Rémi Ronfard et Camélia Guerraoui (2022) dans le cadre de l'article *Blocking notation a tool for annotating and directing theater* proposaient une vision de comment la parole est produite dans ce cadre-là. En effet, selon eux, plusieurs façon de s'exprimer existent : chuchoter, parler, crier, chanter, rire. On peut admettre que cette vision donne une idée de ce qu'est la parole au théâtre, mais elle ne permet pas de savoir comment la parole au théâtre se différencie de la parole en d'autres contextes.

On pourrait cependant penser que cette parole possède des caractéristiques intéressantes et différentes. En effet, le théâtre se veut imiter partiellement un contexte de parole spontanée, il possède notamment beaucoup de dialogues par exemple. Il ne peut cependant être catégorisé comme "parole spontanée" étant donné que les comédiens répètent parfois pendant des mois pour arriver à ce résultat. Si la parole est prise dans cet angle de vue là, on peut estimer que cette dernière est principalement de la parole préparée qui tente d'imiter le spontané. On peut par ailleurs évoquer le théâtre d'improvisation qui fait certainement figure d'exception en termes de préparation de la parole. Les scènes ne sont effectivement pas préparées à l'avance, on pourrait pourtant penser que le registre relève toujours de la parole préparée étant donné le contexte qui reste la scène et le public. Il possède certainement des intonations particulières et une façon de parler spécifique au théâtre.

2. Les outils d'ASR

Alors que l'ASR dans le contexte théâtral reste un champ largement inexploré, il peut sembler pertinent de se tourner vers des approches connexes. En particulier, deux axes méritent une attention particulière : la problématique des langues peu dotées en ressources incluant la notion de variabilité et celle des données non annotées.

Tout d'abord, un enjeu majeur du projet de reconnaissance vocale dédié au théâtre est le déficit de ressources spécifiques à ce domaine. Le théâtre, jusqu'à présent, a été assez peu exploré dans le contexte de l'ASR. Dans l'étude *Automatic Speech Recognition and Query By Example for Creole Languages Documentation* (Macaire et al., 2022), les chercheurs ont examiné des méthodes adaptées à des langues avec des ressources limitées. Ils mettent en évidence des techniques spécifiques pour ces deux langues. Une des techniques clés présentées est une comparaison entre un modèle multilingue et un modèle en français, cette dernière ayant montré une performance supérieure. De plus, l'article indique que l'emploi d'un modèle de langue améliore nettement les résultats comparativement à son absence.

D'autre part, un des obstacles qui peuvent se présenter pour le l'ASR au théâtre est le fait que les données ne sont pas annotées en général et donc potentiellement moins facilement exploitables. Dans l'article *Task Agnostic and Task Specific Self-Supervised Learning from Speech with LeBenchmark* (Evain et al., 2021), les chercheurs présentent un système intégrant de l'ASR sans pour autant que les données ne soient annotées. Le pré-entraînement du modèle a été réalisé sans annotations humaines en utilisant des pseudo-tâches de prédiction. Différentes architectures ont été créées pour différentes tâches, en utilisant des caractéristiques audio spécifiques. Les chercheurs ont évalué les modèles en tant qu'extracteurs de caractéristiques pour ces tâches.

3. Les techniques d'alignement

Les techniques d'alignement applicables au domaine du traitement du langage sont diverses. L'alignement de séquences de texte est essentiel dans divers domaines tels que la bio-informatique (pour aligner des séquences d'ADN), la traduction automatique (pour aligner des textes dans différentes langues). Parmi les méthodes courantes, on trouve Diff et Patch,

des outils traditionnellement utilisés pour détecter et appliquer des différences entre des fichiers, qu'il s'agisse de code source ou de texte général. Une autre méthode courante est la Longest Common Subsequence (LCS), qui identifie la plus longue sous-séquence partagée entre deux séquences. En outre, bien que développés initialement pour des applications en bio-informatique, les algorithmes Smith-Waterman et Needleman-Wunsch peuvent également être exploités pour aligner des séquences de texte.

On peut également recourir à des méthodes plus poussées, incluant de l'apprentissage automatique notamment. C'est le cas des modèles probabilistes tels que les modèles de Markov cachés (HMM, pour Hidden Markov Models), qui peuvent être utilisés pour aligner les séquences entre elles.[2] De plus, des techniques modernes avec des alignements basés sur de l'apprentissage profond existent également. Celles-ci peuvent par ailleurs aligner des séquences entre elles en ne prenant pas uniquement l'aspect graphique des mots en compte mais aussi leur sémantique. Par exemple, les GNN (Graph Neural Networks) sont des types de réseaux de neurones en graphes, et certaines techniques d'alignements utilisent des modèles qui les intègrent.[3]

4. Les données en théâtre français

Le corpus sur lequel j'ai dû travailler durant mon stage n'est pas le seul corpus existant dans le domaine du théâtre français. En effet, en premier lieu, nous pouvons citer le corpus de la Comédie-Française. Cette institution a, à ce jour, enregistré un large panel de pièces entre 1952 et 2021, et les a rendues disponibles sur le site de l'INA aux utilisateurs. Ce sont des pièces de théâtre classique enregistrées par des professionnels. [4] Par ailleurs, le site de l'INA lui-même comporte d'autres pièces de théâtre provenant de diverses autres institutions productrices.

De plus, on peut trouver un corpus d'enregistrements sonores de théâtre sur le site Gallica, la bibliothèque numérique issue de la Bibliothèque Nationale de France (BNF). Ces

[2] Mount D. W. (2009). Using hidden Markov models to align multiple sequences. *Cold Spring Harbor protocols, 2009(7)*, pdb.top41. https://doi.org/10.1101/pdb.top41
[3] Imani, A., Senel, L. K., Jalili Sabet, M., Yvon, F., & Schuetze, H. (2022). Graph Neural Networks for Multiparallel Word Alignment. *Findings of the Association for Computational Linguistics: ACL 2022*, 1384-1396. https://doi.org/10.18653/v1/2022.findings-acl.108
[4] Vodfactory. (n.d.-c). *Streaming illimité de l'INA | madelen*. INA Madelen. https://madelen.ina.fr/home

enregistrements ont été captés entre le XIXème et le XXème siècle et ont été produits par plusieurs comédiens français. Seulement une partie de ces enregistrements est en français. [5]

5. Méthodes d'évaluations pour les systèmes d'ASR

Comme il a été mentionné précédemment, le système va contenir principalement deux éléments : un outil de décodage pour l'ASR et un outil d'alignement de la parole décodée. De ce fait, il peut sembler pertinent de présenter les techniques d'évaluations de ces deux types d'outils. Tout d'abord, pour la partie décodage, une métrique d'évaluation est très courante, il s'agit du WER (Word Error Rate). Il mesure la différence entre les mots reconnus par le système et les mots réels. C'est une métrique pour évaluer la précision de la reconnaissance vocale. Il est la combinaison des erreurs d'insertion, de suppression et de substitution. Il est exprimé en pourcentage, et plusieurs autres métriques découlent de celui-ci en se basant sur la même technique : le SER (Sentence Error Rate) et le PER (Phoneme Error Rate). [6]

D'autre part, des techniques d'évaluation des alignement existent également et certaines d'entre elles sont majoritairement utilisées. On peut citer le taux d'erreur d'alignement, ou AER (Alignement Error Rate), c'est une mesure couramment utilisée pour évaluer les alignements de phrases. Elle combine la précision, autrement dit proportion des alignements proposés qui sont corrects parmi tous les alignements produits (et donc le bruit) et le rappel, la proportion des alignements corrects qui ont été correctement identifiés (et donc le silence). L'idée est que pour avoir un alignement parfait, tous les alignements qui sont "sûrs" (c'est-à-dire certainement corrects) doivent être correctement identifiés. [7] [8]
Le score de similarité entre deux chaînes de caractères qui ont été alignées est une métrique totalement différente. En effet, concentre toute son approche sur la qualité et non la quantité d'alignements. Elle permet d'identifier si l'alignement produit est bon, et mesure en effet la proximité entre les deux chaînes. Plusieurs mesures de similarité existent, on peut citer la

[5] *Voix du théâtre*. (s. d.). Consulté 15 août 2023, à l'adresse https://gallica.bnf.fr/html/und/enregistrements-sonores/voix-du-theatre

[6] He, B., & Radfar, M. (2021). *The Performance Evaluation of Attention-Based Neural ASR under Mixed Speech Input* (arXiv:2108.01245). arXiv. http://arxiv.org/abs/2108.01245

[7] Vilar, D., Popovic, M., & Ney, H. (2006). *AER : Do we need to « improve » our alignments?*

[8] Précision et rappel. (2022). In *Wikipédia*. https://fr.wikipedia.org/w/index.php?title=Pr%C3%A9cision_et_rappel&oldid=194125713

distance de Levenshtein qui mesure la différence entre deux chaînes en se basant sur la présence de substitutions, d'insertions ou d'ajouts.

6. Evaluation des modèles déjà existants

Des modèles tels que Wav2Vec (dans ce cas Wav2Vec2 français) et Whisper sont disponibles en ligne et connaissent un grand succès auprès des personnes qui souhaitent obtenir un décodage de la parole rapidement et efficacement. Bien que ces outils aient fait leurs preuves, leurs taux d'erreurs mots (WER) peuvent connaître des résultats assez élevés.

En effet, si on prend l'exemple de Wav2Vec on constate que les résultats de son WER sont prometteurs pour un certain type de données mais beaucoup moins pour d'autres données. Les modèles wav2vec2.0 préentraînés puis affinés sur des données transcrites. Pour le corpus de parole Common Voice, qui contient de la parole lue, le modèle wav2vec2-FR-3K-large obtient un taux d'erreur de 8%, ce qui est relativement bas et assez prometteur. Cependant, lorsqu'on évalue ce même modèle sur de la parole provenant d'enregistrements de télévision, son taux d'erreur atteint les 26%. Cela signifie certainement que lorsque le modèle est soumis à une parole plus spontanée avec des intonations moins monotones, son WER augmente aussitôt. [9]

Par ailleurs, pour ce qui est du WER de Whisper sur le français il atteint quasiment les 37% dans un environnement non bruité pour le corpus de parole GRACE. Ce corpus contient des conférences TedX, de la parole extraite de vidéos YouTube et des données VoxForge provenant de systèmes de reconnaissance de la parole et qui sont disponibles en open source. Pour ce qui est de ses résultats avec CommonVoice ils sont de 22% environ, et se situent parmi les pires en comparaison avec d'autres langages. Cependant, pour LibriSpeech qui est également de la parole lue le taux d'erreur est de 8% ce qui est un résultat similaire à celui de Common Voice. [10]

[9] Le, H., Alisamir, S., Dinarelli, M., Ringeval, F., Evain, S., & et al. (2022). LeBenchmark, un référentiel d'évaluation pour le français oral *. *34e Journées d'étude sur la parole JEP 2022*, île de Noirmoutier, France.
[10] Vásquez-Correa, J. C., & Álvarez, A. (2023). Novel Speech Recognition Systems Applied to Forensics within Child Exploitation: Wav2vec2.0 vs. Whisper. *Sensors*, *23*(4), 1843. https://doi.org/10.3390/s23041843

De ce fait, étant donnée que la parole du théâtre est principalement préparée, on pourrait penser que les taux d'erreurs sur des modèles comme Whisper ou Wav2Vec se situent à mi-chemin entre ceux de la parole lue et ceux d'environnements plus bruyants intégrants plus de spontanéité au discours.

Partie 3

-

Méthodologie

Pour commencer, le matériel sur lequel je devais travailler est la base de toute la réflexion qui a ensuite été menée pour créer le système le plus approprié à ces données. La notion de reproductibilité et de réutilisation devait cependant rester un objectif. Le corpus de pièces de théâtre provient de Rémi Ronfard, qui les a lui-même obtenues de différentes sources. Ces enregistrements sont des représentations de la pièce de Marivaux *L'île aux esclaves* jouée dans des contextes divers entre 2006 et 2020. En effet, aussi bien dans les conditions d'enregistrement que dans le statut des acteurs, des changements sont notables d'un enregistrement à un autre. Les pièces peuvent être jouées par des amateurs ou par des professionnels et les enregistrements peuvent avoir été captés avec des outils permettant une qualité optimale ou non. Ces facteurs influencent la qualité des enregistrements ainsi que l'éloignement du script de la pièce, les pièces jouées par des professionnels ont tendance à être moins fidèles au texte original et à prendre plus de libertés. Cette hétérogénéité du corpus pourrait permettre de construire un système adapté à plus de données et donc peut-être plus robuste.

Voici ci-dessous un tableau rendant compte de toutes les pièces qui ont été transmises par Rémi.

Metteur en scène	**Année d'enregistrement**
Irina Brook	2006
Axel Joucla	2009
Gilles Droulez	2012
Sébastien Biessy	2015
Gilles Droulez	2016
Jean-Thomas Bouillaguet	2016
Gerold Schumann	2017
Jacques Vincey	2019
Sébastien Biessy	2020

Tableau 1. Corpus des représentations de l'Île des esclaves avec les metteurs en scène et l'année d'enregistrement.

Le choix des représentations que je devais traiter s'est porté sur trois représentations qui semblaient les plus pertinentes. Elles permettent d'avoir un extrait de corpus qui contient aussi bien des représentations de pièces jouées par des amateurs que des pièces avec des professionnels.

D'après mes observations et mes nombreux visionnage des pièces, la pièce de Schumann est celle qui est jouée par des amateurs et donc celle qui se rapproche le plus du script. Quant aux deux autres, celle de Brook et de Vincey, elles auront plus tendance à s'éloigner du script, contenant des extraits qui proviennent totalement de l'écriture du metteur en scène. Ces différences permettent d'avoir une évaluation plus globale de notre système.

Chapitre 1. Traitements préalables

Afin de pouvoir évaluer le système, il a été nécessaire d'annoter les pièces. Nous avons centré notre système sur trois pièces, c'est donc celles-ci que j'ai dû traiter : la pièce de Brook de 2006, la pièce de Schumann de 2017 et enfin la pièce de Vincey de 2019. L'annotation a été faite sur le logiciel ELAN.[11] Il s'agit d'un logiciel disponible en libre accès sur ce site https://archive.mpi.nl/tla/elan/download/ développé dans le cadre du projet Language Archiving par par le groupe technique de l'Institut Max-Planck de psycholinguistique au Pays-Bas. J'ai donc annoté ces pièces par tirades. Pour chaque tirade que je détectais, même s' il y avait des suppressions et des substitutions de mots, j'annotais avec la tirade du texte de théâtre originel en l'insérant dans la fenêtre temporelle que j'avais entendue. La précision de l'annotation se basait donc sur une confiance du travail que j'ai fourni. J'ai ensuite dû extraire les données d'annotations dans le format disponible sur ELAN, que j'ai estimé le plus simple d'utilisation. Il s'agissait du format Quicktime qui est constitué de du timing de début, de la phrase prononcée et du timing de fin. Voici ci-dessous un exemple plus concret d'une tirade dans ce format-là.

[11] *ELAN | The Language Archive*. (2023). https://archive.mpi.nl/tla/elan

[00:03:02.559]
{textEncoding:256}Arlequin ?
[00:03:03.722]

Figure 3. Extrait d'un fichier Quicktime

J'ai ensuite dû convertir les formats qui sont des .srt dans des formats json pour en faciliter le traitement. Pour cela j'ai dû créer un script Python qui traite ces formats, en utilisant la bibliothèque Pysrt qui est disponible en libre accès sur le site https://github.com/byroot/pysrt.[12]

Chapitre 2. Systèmes utilisés ou testés

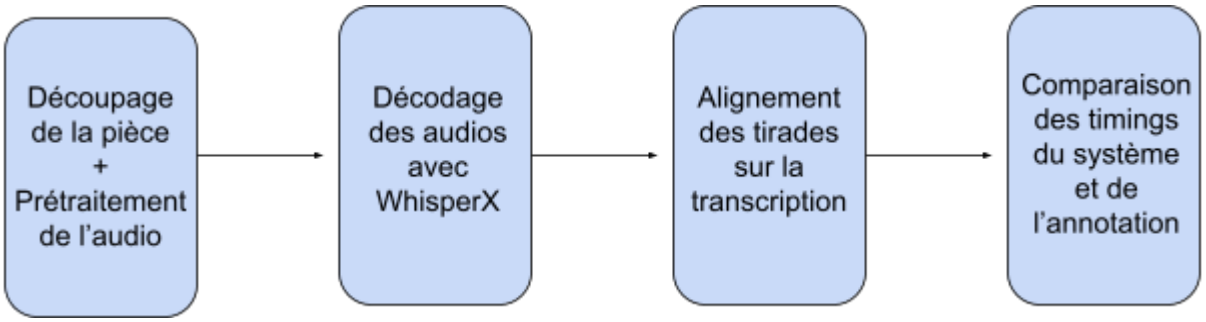


Figure 4. Schéma de l'architecture globale du système

1. Pour l'ASR

1.1. Whisper

Whisper est un modèle développé par l'entreprise OpenAI disponible en open-source sur le github de cette entreprise https://github.com/openai/whisper. Il a été rendu disponible au public en septembre 2022. Il offre la capacité de transcrire dans diverses langues et de traduire ces transcriptions vers l'anglais. De plus, il est compatible avec les formats tels que : m4a, mp3, mp4, mpeg, mpga, wav, et webm.

[12] Mali, Y., Malikwade, V., Kamble, R., & Patil, H. (2022). Smart video summarization using subtitles. *International Research Journal of Modernization in Engineering Technology and Science*, 4(7).

Whisper est conçu autour d'une architecture intégrale, de bout en bout, exploitant le mécanisme des Transformers à la fois en tant qu'encodeur et décodeur. [13] Le schéma numéro 1, que l'on peut retrouver sur le site officiel de Whisper, illustre clairement les différentes étapes de traitement. Lorsqu'un fichier audio d'une durée de 30 secondes est introduit, il est d'abord converti en un spectrogramme log-Mel. Ce spectrogramme est ensuite transmis à l'encodeur. De l'autre côté de l'architecture, le décodeur est formé non seulement pour anticiper et reproduire le texte associé, mais également pour mener à bien diverses autres missions. Parmi ces tâches, on compte la reconnaissance de la langue parlée, la transcription vocale dans différentes langues et même la traduction de la parole vers l'anglais, pour n'en nommer que quelques-unes.

Il est essentiel de souligner la robustesse et la polyvalence de Whisper, et cela découle en grande partie de la manière dont il a été formé. Ayant été entraîné sur un volume impressionnant de 680 000 heures de données multilingues, dont une majorité écrasante (65%) est en anglais, sa performance globale est remarquable. Cette diversité linguistique contribue grandement à son efficacité générale. De plus, la variété des types de données, toutes extraites du web, renforce sa capacité d'adaptation à différents contextes et situations. Cette polyvalence explique en grande partie la robustesse intrinsèque du système. Toutefois, bien que Whisper excelle dans une approche généraliste de la reconnaissance vocale, il ne pourrait pas égaler la précision d'un système conçu et formé pour une spécialisation spécifique ou un domaine défini.

[13] Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2022). *Robust speech recognition via Large-Scale Weak Supervision. arXiv* (Cornell University). https://doi.org/10.48550/arxiv.2212.04356

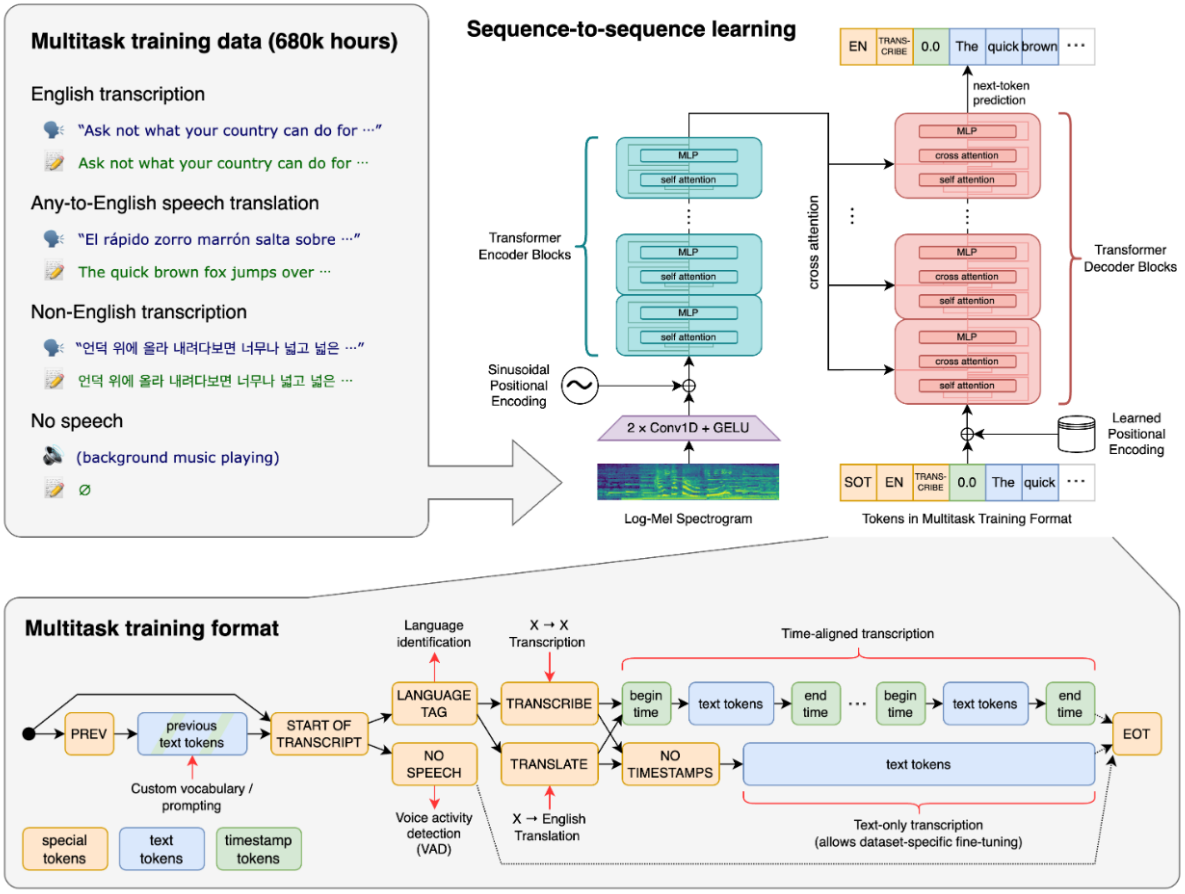


Figure 5. Schéma de l'architecture end-to-end de Whisper provenant du site de Whisper (2022). Récupéré de https://github.com/openai/whisper

1.2 WhisperX

Whisper X utilise comme base Whisper et a ajouté à celui-ci un ensemble de modèles pour permettre la détection de timecodes en plus des transcriptions produites par Whisper.

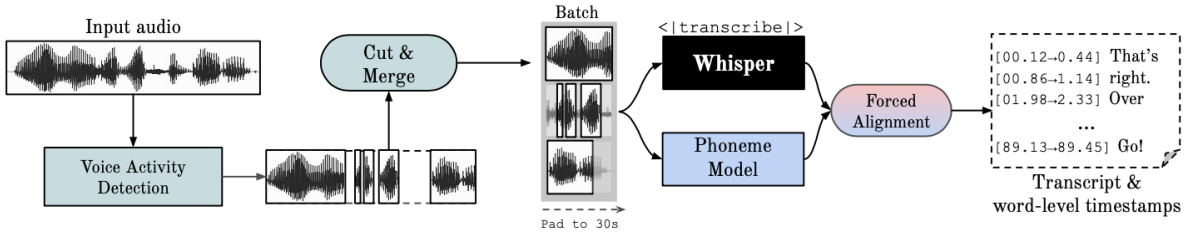


Figure 6. Schéma de l'architecture de WhisperX, provenant du site Github de WhisperX (n. d.)
Récupéré de https://github.com/m-bain/whisperX

Il est a été développé et est régulièrement mis à jour par un membre d'un groupe de recherche à l'Université d' Oxford, le groupe VGG. Il incorpore plusieurs fonctionnalités avancées pour améliorer la précision de la transcription et pour offrir la valeur ajoutée des timecodes.[14]

Premièrement, WhisperX utilise la Détection d'Activité Vocale (VAD). Cette technologie identifie les segments d'un fichier audio qui contiennent de la parole humaine. Elle segmente donc l'audio en détectant les présences ou absences de discours, optimisant ainsi la précision de la transcription. Les segments ainsi identifiés sont ensuite découpés et fusionnés en fenêtres d'environ 30 secondes. Les frontières de ces fenêtres sont définies là où la probabilité de présence de parole est faible. Cette étape présente un avantage notable : elle permet d'utiliser Whisper pour des transcriptions par lots, augmentant ainsi les performances et réduisant les risques de décalage ou d'erreurs dans la transcription.

La dernière étape est celle de l'alignement forcé. À ce stade, WhisperX emploie un modèle basé sur les phonèmes pour aligner la transcription générée avec l'audio. L'ASR basé sur les phonèmes a pour but de reconnaître ces derniers dans la parole. Les phonèmes sont les plus petits segments phoniques perçus dans la représentation mentale des locuteurs. On peut prendre l'exemple de l'élément "g" dans "grand".[15] Cette opération post-traitement est cruciale, car elle aligne la transcription avec les timecodes audio au niveau des mots, offrant ainsi une synchronisation précise.

[14] Bain, M., Huh, J., Han, T., & Zisserman, A. (2023). WhisperX: Time-Accurate Speech Transcription of Long-Form Audio. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2303.00747
[15] *PHONÈME : Définition de PHONÈME*. (2012). Consulté 15 août 2023, à l'adresse https://www.cnrtl.fr/lexicographie/phon%C3%A8me

2. Pour l’alignement

2.1. Biopython/Swalign

Ces deux librairies fonctionnent de façon similaire étant donné qu’elles sont toutes deux basées sur l’algorithme Smith-Waterman, lui-même étant un dérivé d’un autre algorithme d’alignement nommé Needleman-Wunsch. Il s’agit d’une technique prévue pour être utilisée pour des alignements de séquences ADN, donc pour de la biologie et non du traitement du langage.[16] Cependant, nous avons constaté que cette technique pouvait également s’appliquer à du traitement du langage lorsque le texte de la transcription ne s’éloigne pas trop du texte de théâtre original.

Voici comment il fonctionne étape par étape :

- Création d'une matrice de scores et implémentation de celle-ci :

Une matrice de score est créée, en fonction de la taille des séquences à aligner, celle-ci est initialisée à 0.

- Calcul des score de correspondances :

Un score de correspondance est calculé pour chaque position dans la matrice, il se base sur la correspondances des caractères des séquences entre elles. Soit le score est de 1 et on a une correspondance et est positif, soit c'est l'inverse et il est à 0.

- Remplissage de la matrice de score :

La matrice des scores est ensuite remplie de façon itérative, celle-ci est remplie en fonction de l'influence des scores précédents sur l'actuel. (règle de récurrence spécifique)

- Recherche du score maximal et alignement optimal :

On recherche le meilleur score dans la matrice et on recherche l'alignement optimal.

[16] Smith–Waterman algorithm. (2023). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Smith%E2%80%93Waterman_algorithm&oldid=1171675393

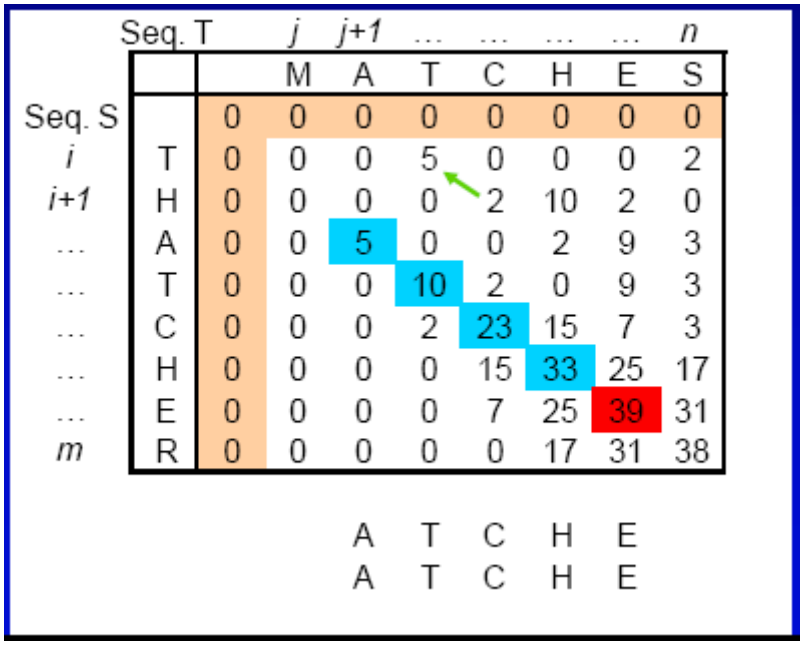


Figure 7. Représentation de l'algorithme d'alignement Smith-Waterman provenant de l'article Pairwise Sequence Alignment and Substitution Matrices par Hala Iqbal (2007).

Chapitre 3. Expérimentations et mise en place du système

Dans les expérimentations que j'ai menées, l'objectif principal était clairement expliqué, cependant, la façon dont nous souhaitions procéder n'était pas prédéfinie. En effet, nous avions une idée de la technique d'approche dans son ensemble qui consiste à faire un décodage puis un alignement, mais les détails de cette technique-là n'étaient pas connus. J'ai dû faire des expériences au fur et à mesure du stage et réajuster par la suite lorsque les résultats n'étaient pas concluants ou qu'un outil ne fonctionnait pas.

1. Premiers tests avec Whisper

Pour commencer, j'ai dû installer Whisper pour le serveur en me rendant sur cette page Web https://github.com/openai/whisper pour me documenter sur son utilisation. L'interface est simple d'usage et accessible à un large panel d'utilisateurs. L'installation se fait avec une première commande de ce type :

```
pin install -U openai-whisper
```

Il fallait ensuite installer FFmpeg qui est en fait une toolbox utilisée pour le prétraitement du signal. J'ai installé mes outils dans des environnements Conda pour des raisons techniques et pratiques. Cela a été fait en lançant les deux commandes suivantes à la suite.

```
conda config --add channels conda-forge
conda install ffmpeg
```

Whisper a ensuite été utilisé par le biais de la commande sur un fichier .wav pour le tester de cette façon là :

```
whisper file.wav
```

Le site de Whisper explique assez clairement comment utiliser Whisper pour un décodage soit dans un script Python soit dans une commande dans la console ou sur le même principe dans un script en langage Bash incluant ces commandes.

Les premier essais ont donc été faits en se basant sur la deuxième option : lancer un décodage simple en ligne de commande. Suite à ça, ce sont les scripts Bash qui ont été utilisés pour lancer le décodage sur un ensemble de fichiers wav issus d'un même fichier wav découpé en segments de 30 secondes adaptés à Whisper.

Les résultats étaient partiellement bons, en effet, le principal problème ressortant de ceux-ci était la langue décodée. Pour des pièces en français uniquement, Whisper détectait d'autres langues telles que du russe ou du coréen. C'est pourquoi il a fallu diriger le système et ainsi le forcer à décoder du français uniquement en le précisant dans la ligne de commande. Cette technique contient certains défauts notamment à cause du fait que pour la pièce de Brook par exemple, une partie de la pièce est en anglais. Ainsi, forcer le système à décoder en français dans ce cas précis peut sembler insensé, mais il s'agit là de la seule solution alternative.

2. De Whisper à WhisperX

La nécessité d'accéder aux timecodes de chaque mot dans la transcription s'est ensuite présentée. Suite à certaines recherches et suggestions, WhisperX s'est avéré être la solution pour pallier ce problème. En effet, il permet d'accéder aux timecodes de chaque mot ce que la version originale de Whisper ne permet pas. L'installation de WhisperX était expliquée étape par étape de façon claire et concise sur le site, cependant elle s'est avérée plus difficile à mettre en place que ce qui était prévu. Lorsque WhisperX a été installé sur le serveur, une simple boucle de commandes dans un script bash a été nécessaire pour réaliser le décodage.

3. Choix de l'outil d'alignement

Les librairies d'alignement disponibles en open source pour Python sont nombreuses, c'est pourquoi il a fallu se tourner vers la solution la plus prometteuse en testant progressivement les outils disponibles en ligne.

Les premiers tests ont été réalisés avec l'outil Swalign, cependant les résultats du script n'étaient pas concluants, c'est pourquoi j'ai dû effectuer d'autres tests avec des librairies différentes. Ensuite, j'ai également testé Difflib sous la recommandation de mon co-encadrant de stage. La librairie Difflib est déjà pré-installée sous Python, malheureusement, dans ce cas-là, les résultats n'étaient pas probants non plus.

La librairie Biopython a été celle qui au final correspondait le mieux aux attentes de résultats, elle utilise le même système que Swalign qui a déjà été expliqué précédemment (Smith-Waterman). Le partitionnement a été fait en tirade pour faciliter la tâche, en effet cela s'avérait être le partitionnement le plus adapté compte tenu du fait que l'annotation préalable issue du logiciel ELAN est découpée en tirade également.

4. Architecture en détails et formats de fichiers

L'architecture des fichiers de traitement du système est composée de scripts Bash et de scripts Python, à cela viennent s'ajouter des lignes de commande qui devront être lancées

directement dans la commande. Comme on peut le constater sur le schéma, l'architecture du système comporte en tout 8 commandes ou scripts à lancer, dont 6 scripts pythons, un script bash et une commande simple. Des bibliothèques Python ainsi que des outils pour la commande doivent être installés pour utiliser le système.

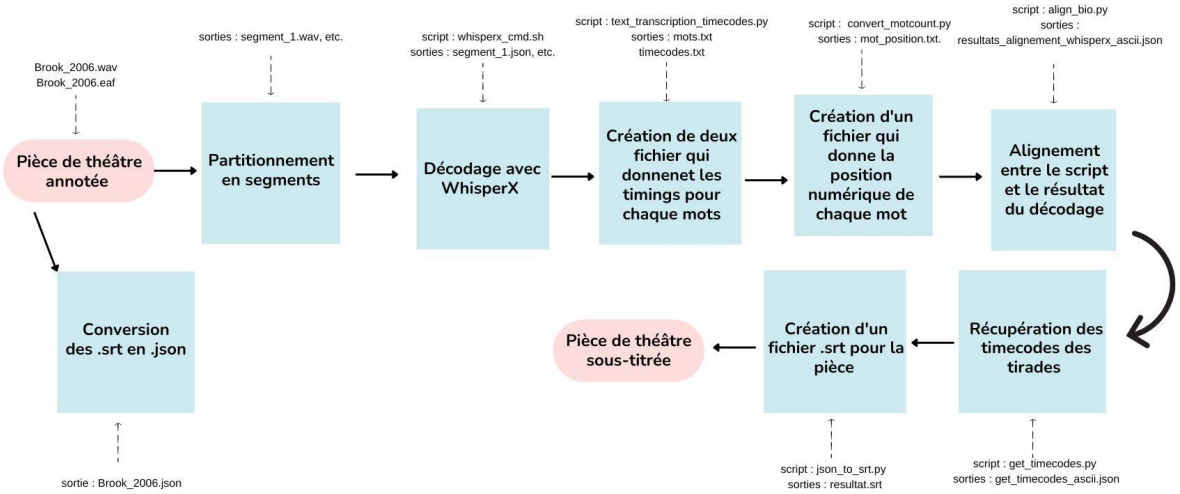


Figure 8. Architecture globale du système d'alignement

Partie 4

-

Evaluation et résultats

1. Méthodes d'évaluation

Le choix de la méthodologie d'évaluation de notre système a été fait conjointement avec Rémi et Benjamin. Suite à diverses réflexions et diverses pistes possibles, nous sommes parvenus à trouver une métrique qui réunissaient tous nos critères d'évaluation. En effet, nous nous basions sur deux aspects capitaux.

Tout d'abord, le recouvrement du timing des tirades était un aspect à prendre en considération. En effet, pour une tirade donnée, si son timing détecté entre dans la fenêtre du timing de référence, le recouvrement est total et donc cela doit être valorisé dans les résultats. En opposition, si le timing détecté n'entre pas ou pas totalement dans la fenêtre temporelle du timing de référence, cela doit être pénalisé dans le résultat.

D'un autre côté, il fallait également prendre en compte le fait que même si le recouvrement était optimal, s'il y avait des timing qui "débordaient" du recouvrement optimal, des pénalités devraient être mises en place. En d'autres termes, la mesure se doit de prendre en compte la précision du timing.

Voici quelques exemples illustratifs des résultats possibles de notre système, la ligne rouge représente le timing détecté et la ligne noire le timing de référence :

Exemple 1

Exemple 2

Exemple 3

Exemple 4

Figure 9. Exemples de résultats de matches entre le timing de référence et le timing matché

Ces quatre exemples donnent un aperçu des résultats possibles du système pour une tirade donnée évoqué lors des réunions. Dans le premier cas, le recouvrement est assuré dans la totalité mais le timing détecté dépasse le timing de référence, dans le cas suivant, le système d'évaluation doit pouvoir pénaliser cela, tout en valorisant le bon recouvrement. Dans le deuxième cas le recouvrement n'est pas total mais il n'y a pas de dépassement du timing, donc pas de pénalité en ce sens-là. Le 3ème exemple combine deux problématiques des deux premiers exemples, en effet il y a un dépassement du timing détecté par rapport au timing de référence et le timing détecté n'assure pas le recouvrement dans la totalité. Enfin, le 4ème exemple est le pire des cas possibles : le timing détecté ne recouvre pas du tout le timing de référence et est totalement en dehors de celui-ci.

C'est en prenant en compte toutes les cas possibles, ainsi que les paramètres les plus importants qu'une métrique s'est avérée plus pertinente que d'autres. Nous pensions devoir la créer de toutes pièces mais la métrique IoU (intersection over union) existe déjà et est déjà utilisée pour des images et plus précisément l'alignement de celles-ci. Ce qui diffère dans notre cas est la dimension : nous avons une dimension en 1D alors que les images sont en deux dimensions.

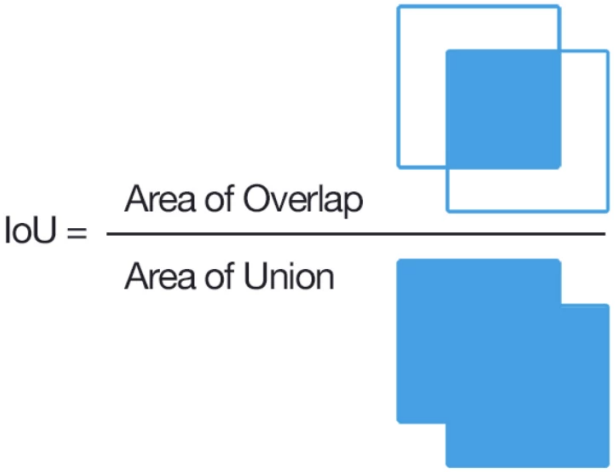


Figure 10. Représentation du calcul de l'IoU, provenant du site https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/

Voici un schéma explicatif illustrant le fonctionnement de la métrique. Pour la calculer, on détermine d'abord l'intersection entre la vérité terrain (timing de référence) et la détection (timing détecté). Ensuite, on évalue l'union de ces deux éléments. La métrique est obtenue en divisant l'intersection par l'union.

Figure 11. Exemples de calculs de l'IoU pour différentes données, provenant du site https://hasty.ai/docs/mp-wiki/metrics/iou-intersection-over-union

Cet exemple est réalisé pour des boîtes mais il permet d'avoir une idée du type de résultats qui pourraient être obtenus pour des données telles que des timings.

2. Résultats

Le processus de sous-titrage des pièces s'est avéré être un défi. Bien que visuellement le sous-titrage semble parfois correct, des métriques spécifiques révèlent des imperfections.

Les résultats des calculs de métriques, en particulier l'IoU, ne sont pas concluants. Un IoU proche de 0 indique des performances médiocres, et, malheureusement, c'est ce que montrent les données pour chacune des pièces étudiées. Le tableau suivant présente une comparaison des valeurs des IoU moyennes et des ratios moyens des chevauchements.

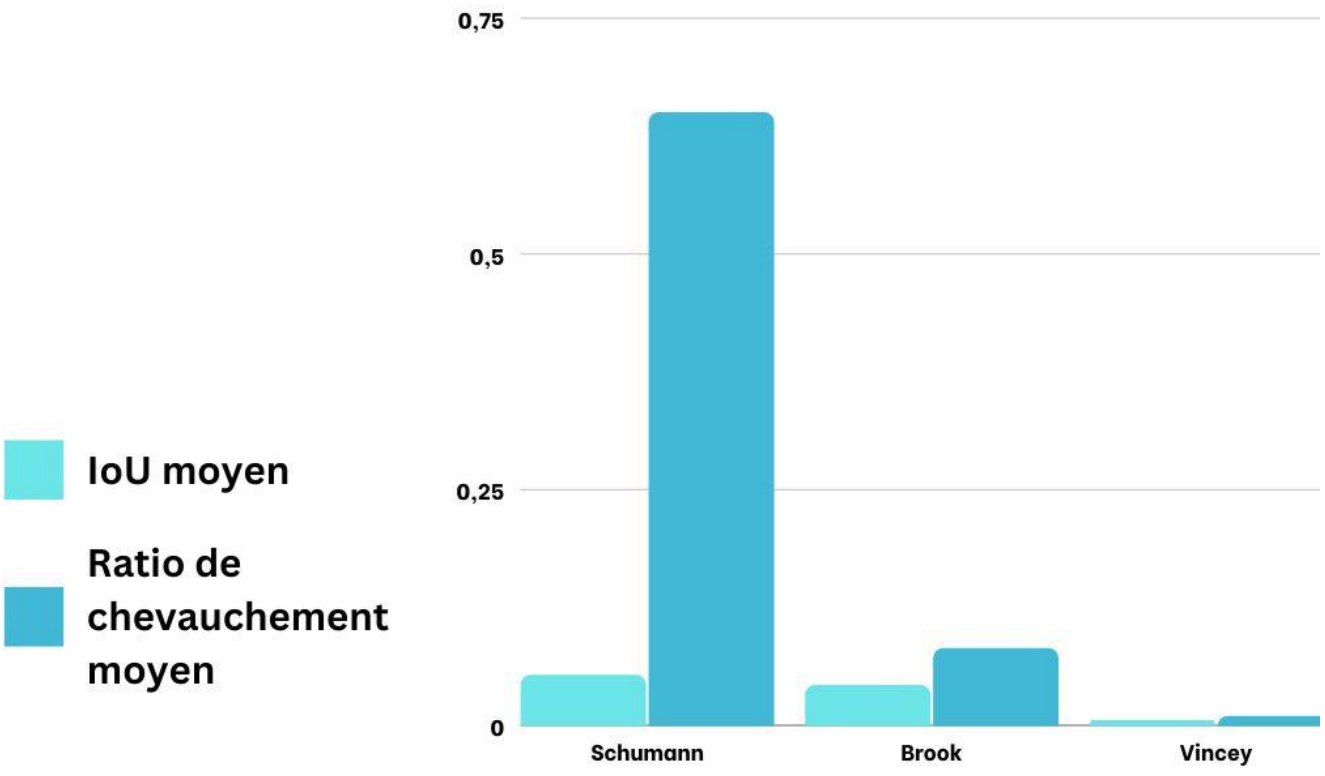


Figure 12. Histogramme en bâton représentant le IoU et le ratio de chevauchement moyens pour chacune des pièces traitées

	Schumann	**Brook**	**Vincey**
IoU moyen	0,0541	0,0430	0.0056
Ratio de chevauchement moyen	0.6497	0.0818	0.0098

Tableau 2. IoU et ratios de chevauchement moyens pour chacune des pièces traitées

Comme on le constate pour les trois pièces les résultats sont autour entre 0,0541 pour Schumann et 0,0056 pour Vincey ce qui est très faible. Certaines tirades ont des temps totalement éloignés des temps de référence. Les résultats de la pièce de Schumann en tant que meilleurs scores étaient plutôt attendus étant donné que c'est cette dernière qui s'éloigne le moins de la pièce de départ, qui s'éloigne le moins du script et où les acteurs et le metteur en scène ont pris le moins de libertés. Cependant, on peut se demander pourquoi le système n'obtient pas de meilleurs scores. Les résultats de la pièce de Brook ne s'éloignent que peu de ceux de la pièce de Schumann (de 0,0111 pour le IoU). Cela est assez inattendu car la pièce possède même des paroles en anglais et on aurait pû penser que le système aurait obtenu un résultat bien pire.

On constate également que pour les scores des ratios moyens des chevauchements (average overlap ratio) ils augmentent pour toutes les pièces par rapport à l'IoU. Il est à noter que le ratio de chevauchement se calcule en divisant le temps de chevauchement par le temps de la tirade, afin d'observer si celui-ci est supérieur à l'IoU. Cela pourrait vouloir dire que lorsqu'une tirade obtient un temps qui matche avec celui de référence, la longueur de la tirade matchée est trop longue par rapport à celle de référence, ce qui ferait baisser le résultat. Pour la pièce de Schumann par ailleurs, le ratio de chevauchement est considérablement augmenté puisqu'il est de 0,6497. Cela signifie que pour plus de la moitié du temps de tirades de référence, le recouvrement est assuré. On peut se pencher un peu plus en détail sur ces résultats afin de mieux les comprendre.

Le tableau ci-dessous présente les moyennes des durées excessives par rapport au temps de référence, ainsi que les moyennes des temps manquants et les moyennes des temps de chevauchement, les chiffres sont exprimés en secondes.

	Schumann	**Brook**	**Vincey**
Durée excessive	12.2610	6.8609	5.1498
Durée manquante	0.6374	4.2641	0.9789

Tableau 3. Durée excessive et durée manquante moyenne pour chacune des pièces traitées

On constate ainsi que les moyennes excessives sont plus élevées dans la globalité par rapport aux durées manquantes, ce qui pourrait potentiellement aller dans le sens de l'hypothèse faite précédemment.

Examinons deux tirades de Schumann pour illustrer ces points :

```
Tirade : Eh ! qui est-ce qui te dit que je ne t'aime plus ?

IOU = 0.2914, Overlap = 1.5810, Overlap Ratio = 1.0000 Excess Duration
= 3.8440, Insufficient Duration = 0.0000
```

Figure 13. Extrait d'un résultat pour une tirade

Cette tirade, matchée dans le texte de Schumann, nous montre un ratio moyen de chevauchement de 1 mais une durée excessive qui fait baisser le IoU. Sur le deuxième extrait issu de Schumann.

```
Tirade : Quel état !

IOU = 0.0000, Overlap = 0.0000, Overlap Ratio = 0.0000, Excess
Duration = 1.6710, Insufficient Duration = 0.0000
```

Figure 14. Extrait d'un résultat pour une tirade

Sur cet extrait on constate que lorsque le IoU et le ratio de chevauchement sont de 0, la durée excessive par rapport au temps de référence est positive. Le système pourrait avoir également une faille dans ce sens là, matchant des timings trop longs par rapport au temps de référence.

Bien que les résultats des calculs de métriques ne soient pas concluants, ils fournissent des indications précieuses. En effet, le fait que les résultats de Schumann soient bien meilleurs en termes de ratio moyen de chevauchement peut orienter l'hypothèse : des interprétations provenant d'acteurs professionnels avec un résultat s'éloignant de la pièce originale pourraient être une des raisons pour laquelle les résultats sont tels qu'ils sont. Un autre aspect qui pourrait s'avérer être un problème est que la partie annotation n'a été faite que par moi, sans validation des annotations par des personnes extérieures. Cela pourrait fausser en partie l'évaluation de notre système.

3. Discussion

Le principe du système pouvait paraître simple dans l'idée de départ. Pourtant, au fur et à mesure de sa conception, des difficultés inattendues se sont présentées. En outre, les résultats finaux n'étaient pas ceux escomptés.

Tout d'abord, l'installation de certaines librairies Python et de toolbox, en particulier WhisperX, a été particulièrement chronophage. J'ai rencontré des difficultés spécifiques de compatibilité entre Conda et WhisperX. C'est finalement grâce à une alternative à Conda, Mamba, que j'ai réussi à faire fonctionner WhisperX. Avec l'aide de collègues du laboratoire, j'ai pu surmonter ces défis plus rapidement. Si d'autres envisageaient de réutiliser mon système, ces problématiques pourraient constituer des obstacles. De plus, l'intervention de mon tuteur de stage a été cruciale pour détecter et rectifier divers bugs dans les scripts.

Ensuite, les résultats de l'évaluation ne sont pas concluants, j'ai tenté d'interpréter ces résultats mais je n'ai pas trouvé de vraie raison à de tels résultats. Le problème est que cela pourrait provenir de plusieurs facteurs, cela pourrait par exemple provenir de WhisperX, de l'alignement ou bien de la combinaison de ces différentes parties du système qui n'est peut-être pas optimale. Il est impossible de répondre à cette question actuellement mais si le système venait à être modifié, ces différentes parties du système devraient être observées et améliorées pour comprendre ces résultats.

De plus, on pourrait également se demander si l'utilisation d'une seule métrique comme le IoU est suffisante pour évaluer le système. En effet, le IoU peut être sensible aux petits décalages dans les tirades, il ne distingue pas la gravité des erreurs, et ne considère pas différents types d'erreurs dans l'alignement. De plus, son interprétation peut ne pas être intuitive, car un faible IoU ne signifie pas nécessairement un mauvais alignement. J'ai tenté d'ajouter des éléments tels que le ratio moyen de chevauchement mais cela n'est certainement pas suffisant. On pourrait se demander si à l'avenir l'utilisation de métriques supplémentaires ne serait pas un apport pour l'évaluation des résultats.

Enfin, il semble important de mentionner la durée de mon stage qui a influencé l'ampleur de l'évaluation et la finesse des ajustements du système. En effet, le temps imparti pour concevoir, développer, tester et évaluer le système était restreint, ce qui a imposé des contraintes quant à la profondeur des tests, la résolution des problèmes identifiés et l'exploration d'approches alternatives. Bien que j'aie fait de mon mieux pour optimiser le système dans ce laps de temps, il est possible qu'avec une période plus prolongée, certains aspects du système auraient pu être améliorés ou approfondis davantage.

Conclusion

L'objectif principal de ce travail était d'aligner avec précision les tirades au théâtre en combinant des outils TAL et informatiques disponibles en ligne et en les adaptant à notre système. Bien que les résultats n'aient pas été aussi satisfaisants que prévu, des informations sont à retirer de cette première tentative concernant diverses implications et complexités inhérentes au système. Cette exploration initiale a mis en lumière les défis spécifiques à relever, ainsi que les domaines potentiels d'amélioration.

J’ai rencontré plusieurs difficultés, allant de l'installation et de la compatibilité des outils à l'interprétation des métriques d'évaluation. Ces défis offrent un aperçu des complexités et des nuances associées à l'alignement précis des tirades, un domaine peu exploré ultérieurement. A l'avenir, il serait intéressant d'explorer des méthodes d'alignement et de reconnaissance de la parole alternatives ou d'affiner les paramètres actuels pour améliorer la précision. De plus, l'ajout de métriques d'évaluation supplémentaires pourrait offrir une meilleure compréhension des domaines spécifiques d'amélioration.

Alors que le monde théâtral continue d'évoluer, la nécessité d'outils d'alignement précis et efficaces ne fait que croître. En continuant à affiner et à développer ces outils, on peut espérer non seulement améliorer notre compréhension des performances théâtrales, mais aussi enrichir l'expérience du public.

Bibliographie

Bain, M., Huh, J., Han, T., & Zisserman, A. (2023). WhisperX: Time-Accurate Speech Transcription of Long-Form Audio. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2303.00747

ELAN | *The Language Archive.* (2023). The Language Archive. https://archive.mpi.nl/tla/elan

Evain, S. et al. (2021). Task Agnostic and Task Specific Self-Supervised Learning from Speech with LeBenchmark.

He, B., & Radfar, M. (2021). The Performance Evaluation of Attention-Based Neural ASR under Mixed Speech Input. *arXiv.* http://arxiv.org/abs/2108.01245

Imani, A., Senel, L. K., Jalili Sabet, M., Yvon, F., & Schuetze, H. (2022). Graph Neural Networks for Multiparallel Word Alignment. *Findings of the Association for Computational Linguistics: ACL 2022,* 1384-1396. https://doi.org/10.18653/v1/2022.findings-acl.108

Katsalis et al. (2022). Employing Speech Recognition Technologies for Improving Accessibility and Augmenting the Theatrical Experience.

Le, H., Alisamir, S., Dinarelli, M., Ringeval, F., Evain, S., & et al. (2022). LeBenchmark, un référentiel d'évaluation pour le français oral. *34e Journées d'étude sur la parole JEP 2022*, île de Noirmoutier, France.

Macaire et al. (2022). Automatic Speech Recognition and Query By Example for Creole Languages Documentation.

Mali, Y., Malikwade, V., Kamble, R., & Patil, H. (2022). Smart video summarization using subtitles. *International Research Journal of Modernization in Engineering Technology and Science,* 4(7).

Mount D. W. (2009). *Using hidden Markov models to align multiple sequences. Cold Spring Harbor protocols*, 2009(7), pdb.top41. https://doi.org/10.1101/pdb.top41

PHONÈME : Définition de PHONÈME. (2012). Consulté le 15 août 2023, à l'adresse https://www.cnrtl.fr/lexicographie/phon%C3%A8me

Précision et rappel. (2022). *In Wikipédia.* https://fr.wikipedia.org/w/index.php?title=Pr%C3%A9cision_et_rappel&oldid=191125713

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2022). Robust speech recognition via Large-Scale Weak Supervision. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2212.04356

Smith–Waterman algorithm. (2023). *In Wikipedia.* https://en.wikipedia.org/w/index.php?title=Smith%E2%80%93Waterman_algorithm&oldid=1171675393

Vásquez-Correa, J. C., & Álvarez, A. (2023). Novel Speech Recognition Systems Applied to Forensics within Child Exploitation: Wav2vec2.0 vs. Whisper. *Sensors*, 23(4), 1843. https://doi.org/10.3390/s23041843

Vilar, D., Popovic, M., & Ney, H. (2006). AER : Do we need to « improve » our alignments? Voix du théâtre. (s. d.). Consulté le 15 août 2023, à l'adresse https://gallica.bnf.fr/html/und/enregistrements-sonores/voix-du-theatre

Vodfactory. (n.d.-c). Streaming illimité de l'INA | madelen. *INA Madelen.* https://madelen.ina.fr/home

Zhou, D., Fang, J., Song, X., Guan, C., Yin, J., Dai, Y., & Yang, R. (2019). IoU Loss for 2D/3D Object Detection. *2019 International Conference on 3D Vision (3DV),* 85-94. https://doi.org/10.1109/3DV.2019.00019

Table des figures

Table des tableaux

Sigles et abréviations utilisées

ASR : Automatic Speech Recognition, en français : reconnaissance automatique de la parole

IoU : Intersection over Union, en français : intersection sur l'union

PER : Phoneme Error Rate, en français : taux d'erreurs des phonèmes

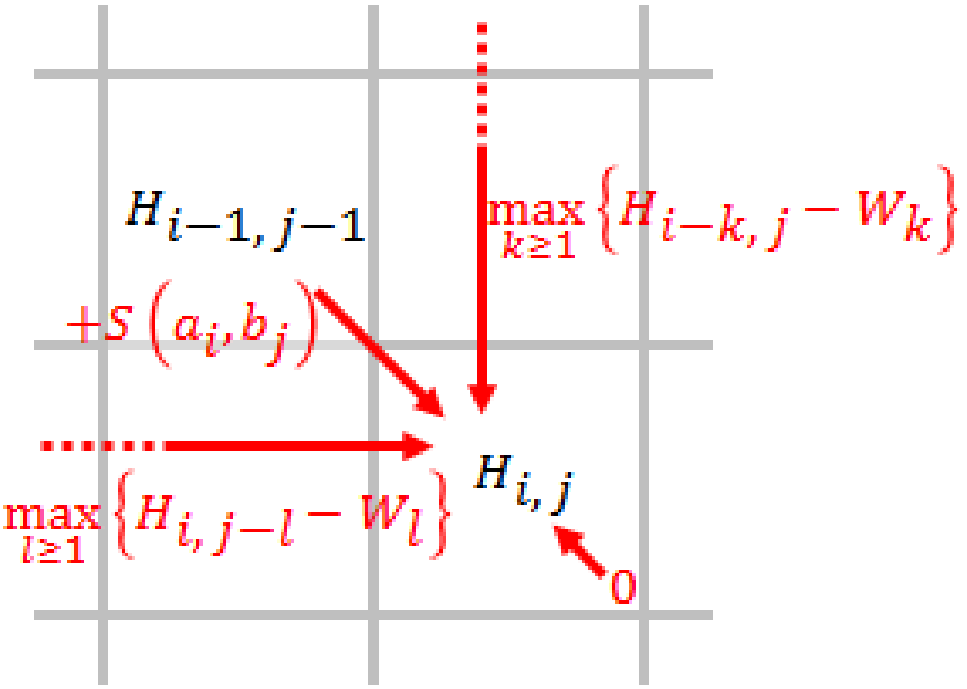
SER : Sentence Error Rate, en français : taux d'erreur des phrases

TAL : Traitement Automatique des Langues

WER : Word Error Rate, en français : taux d'erreurs des mot

Table des annexes

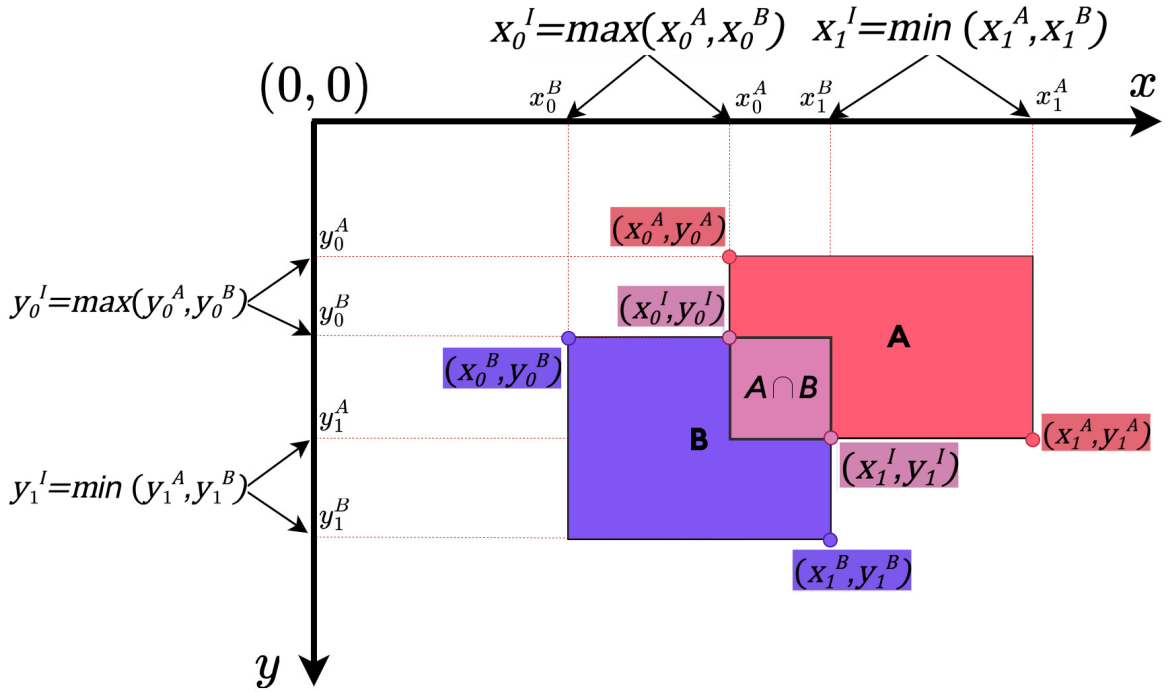
Annexe 1 : Représentation des calculs constituant l'algorithme de Smith-Waterman



Annexe 2 : Méthode de calcul de la matrice des scores

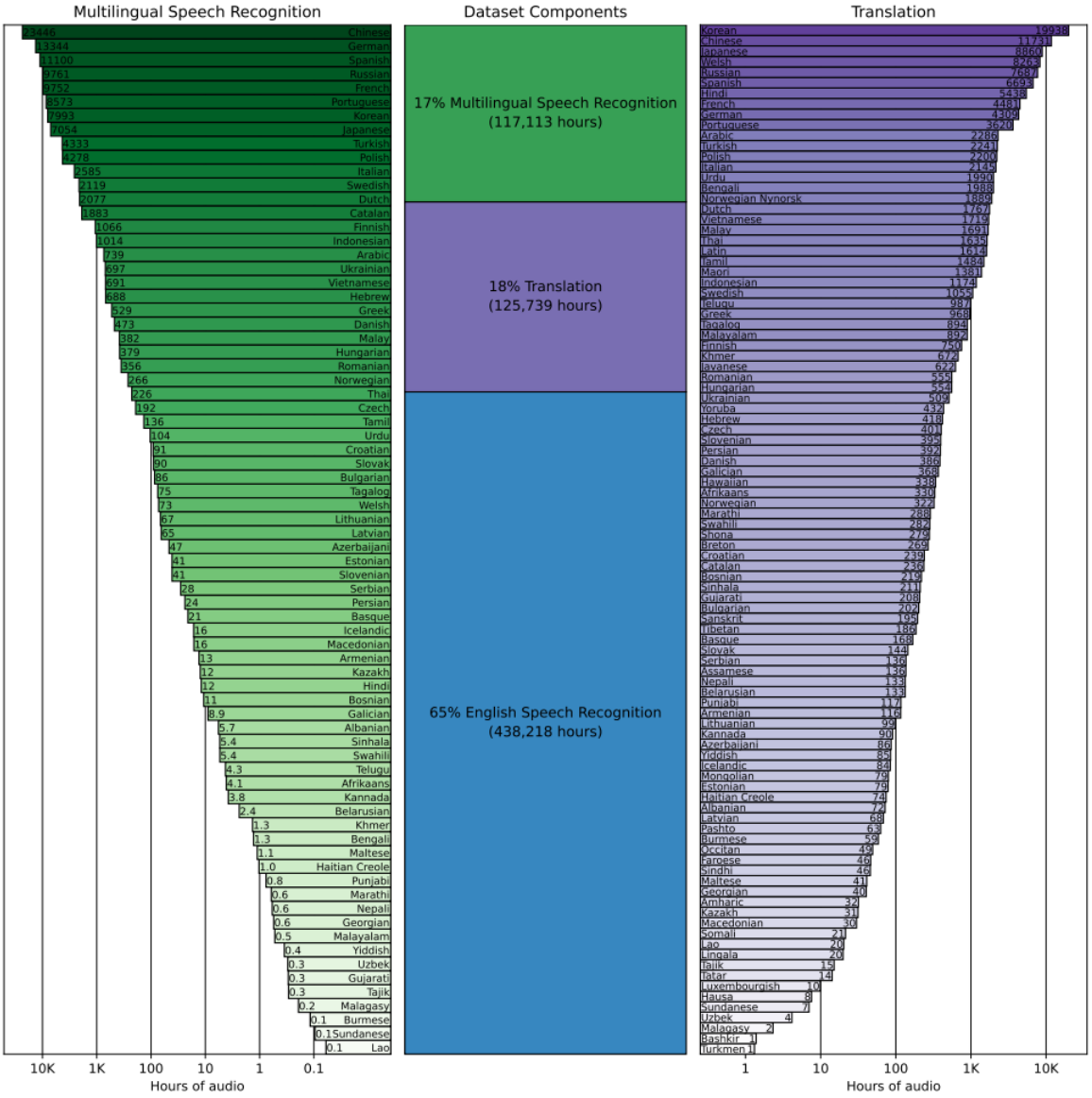
$$M(i,j) = \max \begin{cases} 0 \\ M(i-1,j-1) + D(A_i, B_j) \\ M(i-1,j) + \Delta \\ M(i,j-1) + \Delta \end{cases}$$

Annexe 3 : Méthode de calcul de l'IoU

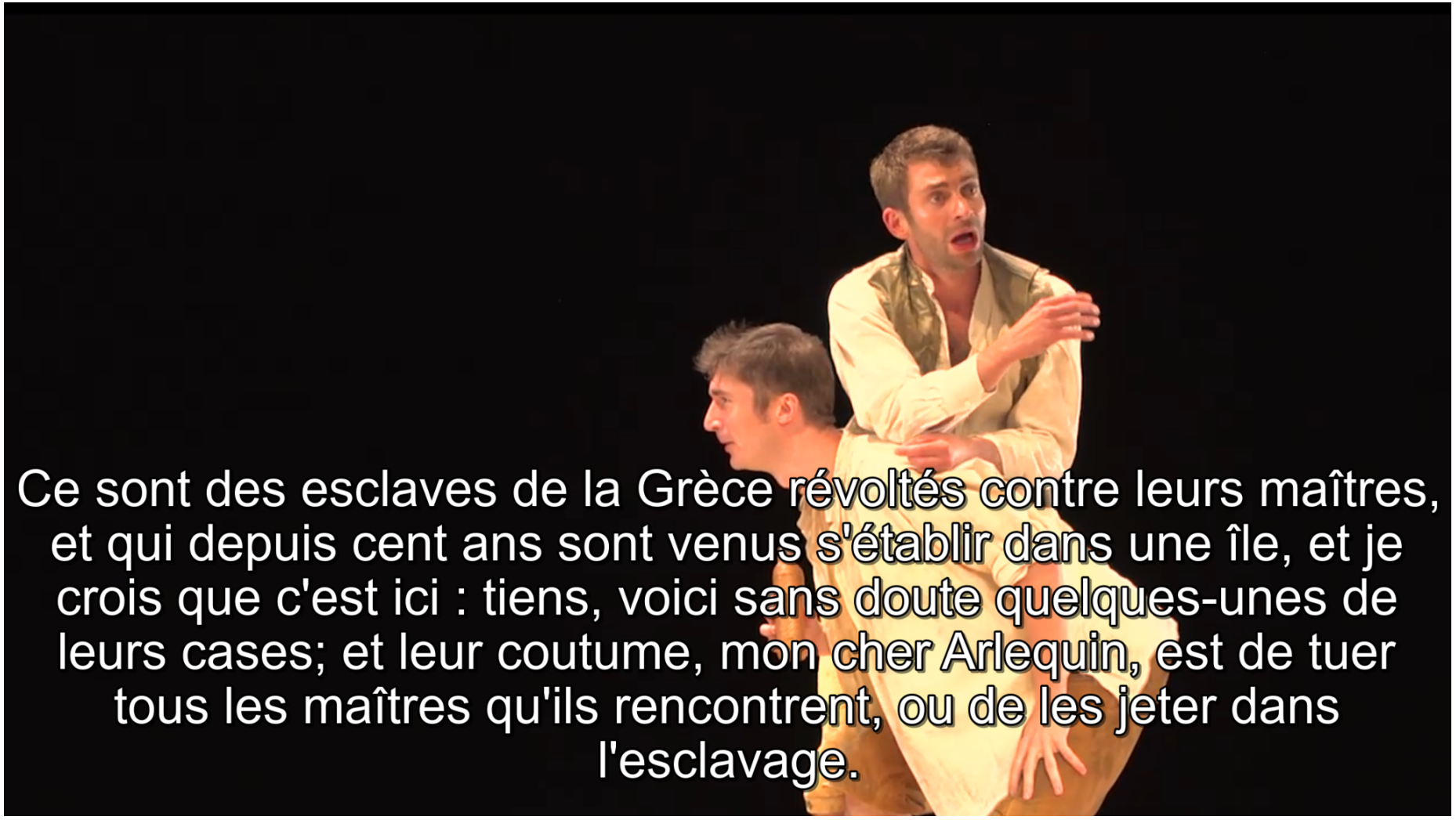


$$Intersection\ over\ Union\ \left(IoU\right) = \frac{|\mathrm{A} \cap \mathrm{B}|}{|\mathrm{A}| \cup |\mathrm{B}|}$$

Annexe 4 : Répartition des jeux de données d'entraînement de Whisper



Annexe 5 : Exemple de sous-titrage pour la pièce de Schumann



Annexe 6 : Exemple d'utilisation du logiciel ELAN pour l'annotation manuelle

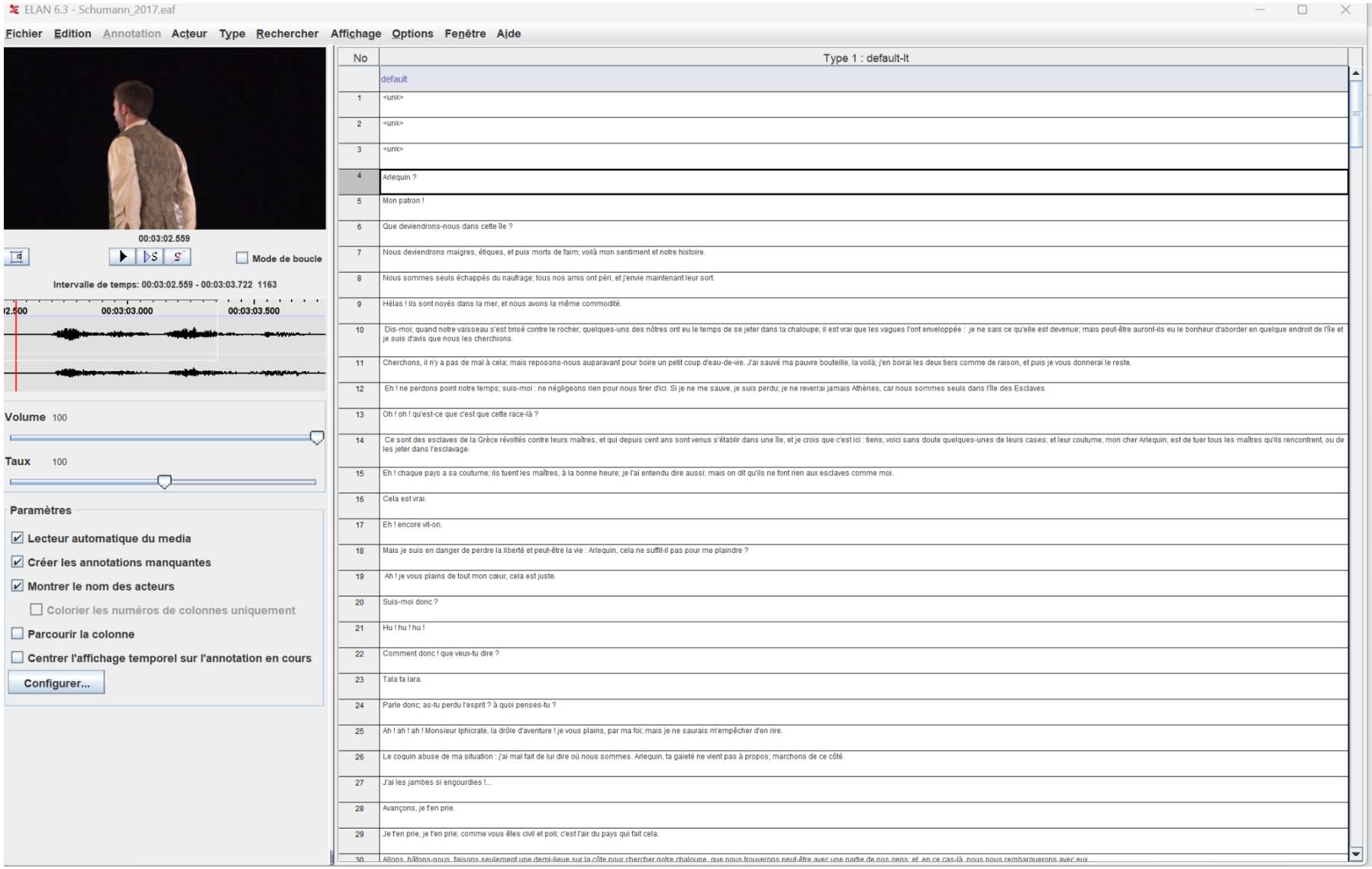


Table des matières

Mots-clés : Reconnaissance automatique de la parole, théâtre, WhisperX, alignement de texte, algorithme de Smith-Waterman

RÉSUMÉ

Le théâtre, avec ses nuances de discours spécifiques, présente des défis particuliers dans le domaine de la reconnaissance automatique de la parole (ASR). Ce rapport détaille la conception d'un système d'ASR adapté spécifiquement au contexte théâtral en s'appuyant sur des outils existants. WhisperX, un outil de reconnaissance de la parole de pointe, a été adapté et optimisé pour cette application. Pour assurer un alignement précis avec le texte de théâtre original, nous avons utilisé l'algorithme Smith-Waterman. Bien que les performances initiales n'aient pas été à la hauteur des attentes, le travail souligne l'importance de poursuivre la recherche dans ce domaine. Plusieurs pistes d'amélioration sont proposées pour d'éventuelles modifications dans le cadre de travaux futurs.

Keywords: Automatic Speech Recognition, theater, WhisperX, text alignment, Smith-Waterman algorithm

ABSTRACT

The theater, with its specific speech nuances, poses unique challenges in the field of Automatic Speech Recognition (ASR). This report details the design of an ASR system specifically tailored for the theatrical context, leveraging existing tools. WhisperX, a state-of-the-art speech recognition tool, was adapted and optimized for this application. To ensure accurate alignment with the original theater text, we employed the Smith-Waterman algorithm. Although the initial performances did not meet expectations, the work underscores the importance of continued research in this area. Several avenues for improvement are proposed for potential refinements in future endeavors.