Rapport projet SAM
Prédiction multimodale du tour de parole en conversation dyadique

Réalisé par Victor Tancrez, Tony Nogneng et Paul Peyssard
Encadré par Leonor Becerra & Eliot Maes
M2 IAAA, AMU

Abstract

Ce rapport présente une étude approfondie sur la prédiction de prise de parole dans les conversations, un aspect crucial des systèmes d'interaction homme-machine. Avoir un modèle efficace sur cette tâche est loin d'être trivial. Nous explorons les complexités du traitement de données multimodales, en nous concentrant spécifiquement sur les techniques de fusion tardive et précoce dans les modèles de deep learning. En tirant parti d'un riche ensemble de données, le corpus paco-cheese, comportant de l'audio, de la vidéo et des données textuelles, notre recherche se penche sur les subtilités de la communication verbale et de la gestuelle du visage. Nous traitons et analysons les données, développons des bases de référence unimodales et construisons des modèles multimodaux pour prédire les événements de prise de parole. Les conclusions de cette étude contribuent non seulement à la compréhension des interactions multimodales, mais mettent également en lumière les défis et les expériences d'apprentissage rencontrés dans la gestion de projets de deep learning.

1 Introduction

Ce projet, mené dans le cadre de l'UE Signal et Apprentissage Multimédia du Master Intelligence Artificielle Apprentissage Automatique à Aix-Marseille Université, se concentre sur la détection de tours de parole au sein de conversations entre deux personnes. Cette tâche, essentielle dans les systèmes de communication et d'interaction homme-machine, consiste à anticiper le moment où un interlocuteur dans une conversation est susceptible de prendre la parole ou de céder la parole à l'autre participant. Pour résoudre cette tâche difficile nous avons exploité des modèles de deep learning multimodaux. En apprentissage automatique, un modèle multimodal fait référence à un système capable de traiter et d'analyser des données issues de multiples sources ou types de données, telles que le texte, l'audio, la vidéo, etc. Ces modèles intègrent et interprètent ces différentes formes de données pour améliorer la compréhension et la performance de la tâche à accomplir. Dans notre cas, ils permettent une analyse plus riche et plus nuancée des conversations. Nous mettons l'accent sur les techniques de fusion de modèles et de données et leur impact sur la performance des systèmes.

Ainsi, notre problématique concerne l'étude et l'implémentation de modèles multimodaux pour résoudre une tâche de prédiction de tour de parole.

Afin de répondre à cette problématique, nous commencerons par une présentation du dataset sur lequel nous avons travaillé et de la tâche précise que nous avons choisie de résoudre. Ensuite, nous présenterons les modèles unimodaux que nous avons initialement implémentés. Par la suite, nous expliquerons comment nous avons utilisé ces modèles unimodaux pour développer un modèle de late fusion, suivi par la création de notre modèle d'early fusion. Pour conclure, nous présenterons nos résultats sur cette tâche.

2 Les Données

Le dataset que nous avons utilisé durant notre projet est le Corpus PACO-cheese, une collection diversifiée de dialogues dyadiques enregistrés. Ce dataset est spécifiquement conçu pour l'étude des interactions verbales et non verbales dans le contexte de la communication humaine. Il comprend des enregistrements audio (WAV, mp4), les vidéos (mp4) correspondants à ces audios ainsi que des transcriptions textuelles (csv), et des métadonnées détaillées telles que les identifiants de locuteur, les timestamps, le contenu des échanges et autres. L'utilisation de PACO-cheese nous permet d'accéder à des données réalistes et variées, essentielles pour l'analyse et la compréhension des dynamiques de tour de parole. Cette richesse de données soutient nos efforts pour développer des modèles de deep learning précis et robustes, capables de détecter et d'analyser efficacement les tours de parole dans des dialogues naturels.

2.1 Métadonnées

Les métadonnées sont stockées dans des fichiers CSV qui contiennent des informations détaillées sur les tours de parole. Dans le CSV que nous avons choisi chaque ligne représente une unité de production individuelle (IPU) avec des colonnes comme `ipu_id`, `speaker`, `start`, `stop`, `turn_after`, et `yield_at_end`. Le champ `speaker` indique le locuteur actuel, tandis que `start` et `stop` fournissent les timestamps de début et de fin de l'IPU. Nous définissons chaque ligne comme étant un segment de parole.

3 Tâche à résoudre

3.1 Choix des labels à prédire

Nous avons décidé d'utiliser les colonnes `turn_after`, et `yield_at_end` des fichiers CSV comme labels à prédire.

Nous cherchons à effectuer cette prédiction pour chaque "ipu_id". Le label `yield_at_end` identifie un instant où la personne qui a la parole cède le tour de parole à son interlocuteur, tandis que le label `turn_after` identifie un instant où le tour de parole change. Le tour de parole n'est pas équivalent au locuteur actuel. La personne qui a le tour de parole est celle qui est en train de transmettre du contenu linguistique. Lorsqu'un individu parle brièvement pour acquiescer et pour signaler qu'il écoute, il devient temporairement le locuteur mais ne prend pas pour autant le tour de parole.

Le problème avec `turn_after`, est qu'une personne peut céder la parole sans que son interlocuteur la prenne, et inversement quelqu'un peut prendre la parole sans que la personne qui avait la parole lui ait cédé. Or, il est naturel pour un humain de reconnaître quand la parole va être cédée à partir des données visuelles et audio des secondes précédentes, mais il est beaucoup plus difficile pour un humain de prédire le moment où quelqu'un va prendre la parole. En effet, cela dépend de données bien plus complexes que le contexte immédiat, tels que la motivation de l'interlocuteur à prendre la parole, la tendance d'une personne à couper plus ou moins la parole, son état émotionnel, le lien social entre les deux interlocuteurs, et de nombreux autres facteurs.

Pour rester concis, nous nous concentrerons sur l'analyse des prédictions pour turn_after.

3.2 Déséquilibre des classes

Quelle que soit la tâche à résoudre, nous avons constaté un important déséquilibre des classes, avec au moins trois fois plus de données qui ne sont pas concernées par le changement de parole. C'est un point que nous avons dû prendre en compte pour apprendre nos modèles.

4 Baselines unimodales

4.1 Modèle à base de texte

Dans cette baseline, nous utilisons exclusivement des données textuelles pour prédire les tours de parole, établissant un point de référence pour les performances avant l'intégration de modalités supplémentaires.

4.1.1 Pertinence des Données Textuelles dans la Prédiction de Tours de Parole

Il semble possible d'apprendre un modèle capable de prédire le changement de parole en utilisant uniquement des données textuelles. En effet, certains éléments présents dans une conversation peuvent indiquer un changement imminent de locuteur. Par exemple, si un locuteur pose une question telle que "Comment vas-tu ?", il est généralement attendu que la parole passe à l'autre personne pour répondre.

4.1.2 Le Traitement des Données Textuelles

Suite au choix des targets et au traitement du CSV, nous avons récupéré les conversations retranscrites et nous avons prétraité ces phrases en utilisant le tokeniseur CamemBERT. Il convertit

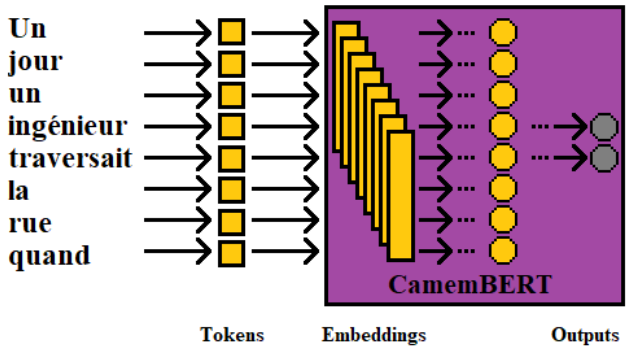

Figure 1: Modèle à base de texte (Image conçue et réalisée par nous-même)

les phrases en séquences de tokens qui sont ensuite données en entrée du modèle.

4.1.3 L'Architecture du Modèle

La Figure 1 décrit l'architecture du modèle. Le modèle est basé sur l'architecture CamemBERT, exploitée pour sa capacité à comprendre le contexte et la structure linguistique du texte. L'architecture se compose principalement d'un encodeur Transformer, capable de traiter les séquences de tokens et d'en extraire des caractéristiques significatives. Grâce à cette représentation de notre séquence de texte en entrée, nous entraînons un modèle à prédire si la séquence de texte correspond ou non à un tour de parole.

Le modèle est entraîné en utilisant une fonction de perte de cross-entropy, avec des poids associés à chaque classe pour prendre en compte le déséquilibre des classes.

4.2 Modèle Audio

4.2.1 Pertinence des Données Audio dans la Prédiction de Tours de Parole

Les caractéristiques audio telles que le ton, le volume, le rythme et les pauses jouent un rôle essentiel dans la dynamique conversationnelle et peuvent être des indicateurs clés pour anticiper un changement de locuteur. Par exemple pour le cas des pauses et des hésitations, elles sont souvent des moments où un changement de tour de parole est possible. Un silence prolongé peut indiquer qu'un locuteur a terminé son intervention ou invite son interlocuteur à prendre la parole.

4.2.2 Les Fichiers Audio

Nous utilisons principalement des fichiers audio au format WAV, situés dans le dossier '2_channels'. Le traitement des données audio implique plusieurs étapes :

1. Identification des fichiers audio : Nous associons chaque segment de parole à son fichier audio correspondant en utilisant le nom des locuteurs et les identifiants de dyade.

2. Extraction de segments audio : À l'aide des timestamps de début et de fin (start et stop) de chaque IPU, nous extrayons les segments audio correspondants du fichier WAV complet.

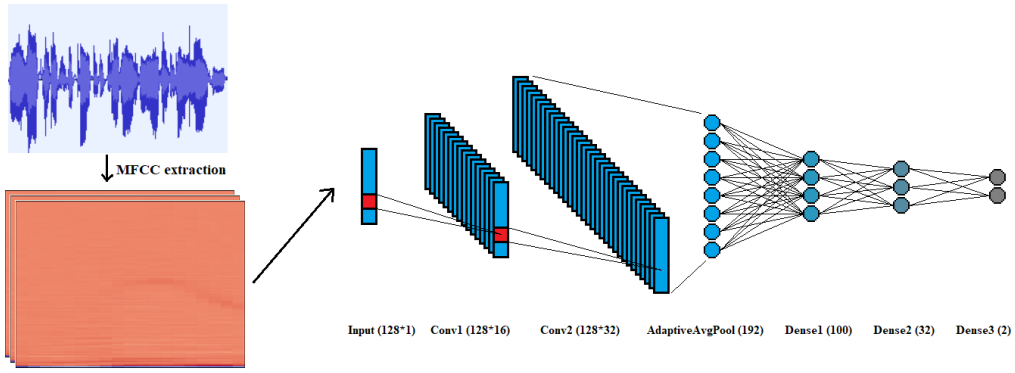

Figure 2: Modèle Audio (Image conçue et réalisée par nous-même)

3. Traitement des segments audio : Chaque segment audio est ensuite traité pour en extraire des caractéristiques. Nous utilisons `librosa` pour calculer les MFCCs (Mel-Frequency Cepstral Coefficients, permettent de capturer les propriétés acoustiques d'un signal sonore en se basant sur une représentation du spectre de puissance sur une échelle de fréquence Mel, plus conforme à la perception humaine.) de chaque segment, qui sont ensuite moyennés pour obtenir un ensemble de caractéristiques représentatif.

Ces processus de traitement des fichiers CSV et audio sont essentiels pour préparer nos données à l'entraînement et à l'évaluation des modèles de détection des tours de parole.

4.2.3 L'architecture du modèle

4.3 Modèle Vidéo

4.3.1 Pertinence des Données Vidéo dans la Prédiction de Tours de Parole

Les données vidéo capturent les mouvements corporels, les expressions faciales, et le langage non verbal, qui sont des éléments clés dans la communication humaine et peuvent fournir des indices importants sur le changement de tour de parole.

Par exemple, les expressions faciales et les mouvements des yeux sont des indicateurs significatifs des intentions de communication. Par exemple, un regard direct vers l'interlocuteur peut signifier une invitation à parler, tandis qu'un regard fuyant peut indiquer une pause ou une fin de prise de parole.

4.3.2 Les Fichiers Vidéos

Pour tenter d'enrichir notre analyse multimodale, nous avons exploré le traitement des fichiers vidéo du corpus PACO-cheese. Le processus d'extraction de caractéristiques des vidéos a été initié, bien que son intégration complète dans nos modèles n'ait pas été possible en raison de contraintes de temps et de ressources. Il nous aurait fallu plusieurs jours d'extractions de features et d'entraînement avec les ressources que nous avions à dispositions.

4.3.3 Extraction et Traitement des Segments Vidéo

Le traitement des fichiers vidéo a commencé par la localisation des fichiers pertinents en fonction des dyades et des locuteurs. Chaque fichier vidéo correspondant à une dyade et un locuteur spécifique était identifié et traité. Des segments vidéo étaient ensuite extraits pour chaque IPU, en utilisant les timestamps de début et de fin fournis dans les données. Ces segments étaient capturés en tenant compte du nombre d'images par seconde (FPS) des vidéos pour assurer une extraction précise.

4.3.4 Prétraitement et Extraction de features

Nous avons premièrement prétraité chaque frame des segments vidéo afin d'utiliser le modèle VGG16, qui est un modèle de réseau de neurones convolutif conçu pour la classification d'images et la détection d'objets. Cependant, nous nous sommes rendu compte que les features extraites n'étaient pas utiles dans ce contexte. Nous avons donc utilisé un autre modèle afin d'extraire les keypoints des visages, qui nous semblent être des features plus adaptées au problème.

Nous avons décidé d'utiliser un modèle basé sur un réseau de neurones convolutifs pré-entrainé (res10_300x300_ssd_iter_140000.caffemodel) et un modèle de landmarks faciaux (lbfmodel.yaml), tout deux importés via OpenCV. Ces modèles sont capables de détecter les visages dans chaque frame et d'identifier des key-points visage, tels que les coins des yeux, de la bouche, le bout du nez, etc., qui sont cruciaux pour comprendre l'expression faciale et l'orientation du visage.

Nous avons testé plusieurs modèles ayant la même fonction mais nous avons constaté que l'extraction de features ne fonctionnait pas toujours correctement. En effet, la variabilité de la position des visages dans les vidéos peut entraver leur détection, notamment lorsqu'ils sont orientés différemment, qu'ils sont partiellement obscurcis (cheveux, ombre, micro, ...) ou encore des angles de vue inhabituels.

L'extraction de features peut également être sensible aux biais présents dans la base de données d'entraînement des modèles. Si les données contiennent majoritairement des visages de certains genres, âges ou ethnies, la généralisation à des situations plus diversifiées peut être d'une qualité moindre

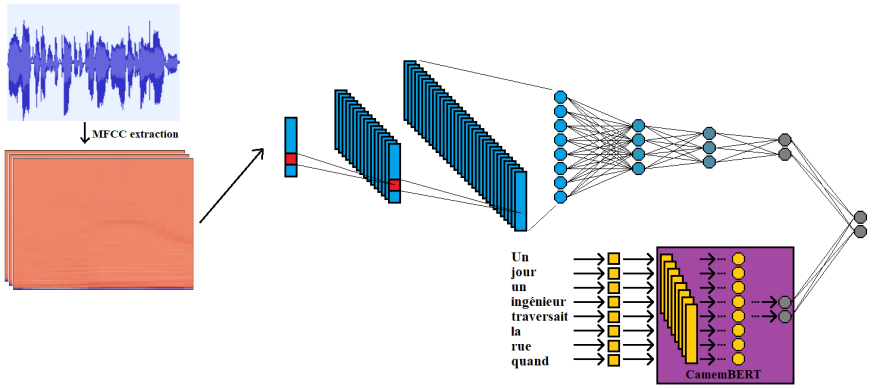

Figure 3: Late Fusion (Image conçue et réalisée par nous-même)

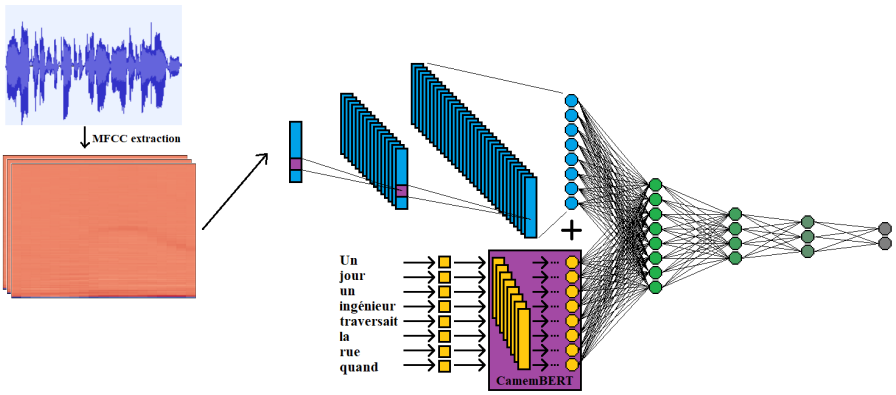

Figure 4: Early Fusion (Image conçue et réalisée par nous-même)

que celle obtenue lors de la validation du modèle.

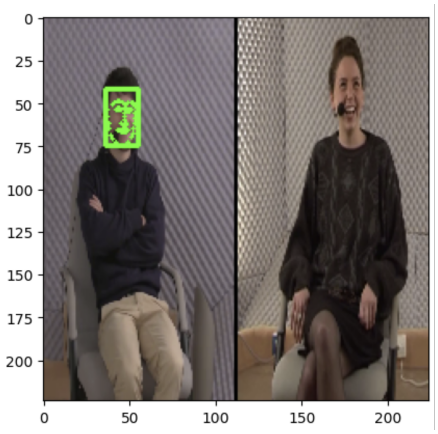

Figure 5: Détection des visages (Image issue de Paco-Cheese)

Dans la Figure 5, nous observons un exemple de ce phénomène : pendant un rire et avec le microphone placé devant la bouche, le visage de la personne à droite n'est pas détecté et les features ne sont pas extraites totalement. Ce type de phénomène peut donc potentiellement diminuer la performance du système.

4.3.5 L'architecture du modèle

Nous avons mis en place un modèle basé sur les réseaux de neurones récurrents, spécifiquement les LSTM (Long Short-Term Memory), pour traiter les séquences temporelles des keypoints des visages extraits des vidéos. Il est important de noter que cette partie de notre projet était exploratoire, notre objectif étant de démontrer la faisabilité de l'intégration des données vidéo dans notre pipeline de traitement multimodal. Le modèle n'a donc pas été entraîné avec l'ensemble des données.

Nous avons volontairement choisi de faire un modèle très basique puisque nous ne disposions pas de suffisamment de ressources pour pouvoir finetuner le modèle. Nous avons donc choisi de nous focaliser sur le texte et l'audio afin d'obtenir les meilleurs résultats possible en early fusion et en late fusion avec les moyens que nous avions.

5 Modèles multimodaux

5.1 Late Fusion

Après avoir entraîné les modèles multimodaux séparément, leurs poids sont figés. La fusion tardive s'effectue par une couche dense qui apprend à prédire les changements de tours de paroles en se basant sur les sorties de chaque modèle unimodal (Figure 4), tirant profit des informations de chaque modalité pour améliorer la prédiction globale. Pour celle-ci, nous avons utilisé les modèles de texte et d'audio.

5.1.1 Intégration des Données

Le corpus d'entraînement pour le modèle de late fusion reste le même que celui des modèles unimodaux.

5.1.2 Fusion des Modèles

La stratégie repose sur la combinaison des modèles unimodaux. Chaque modèle unimodal fournit une prédiction liée à sa modalité, puis ces prédictions sont concaténées dans un vecteur. À partir de ce vecteur, une couche dense réalise la prédiction finale du modèle, en apprenant la pondération optimale des sorties des modèles unimodaux.

5.2 Early Fusion

L'early fusion combine des caractéristiques unimodales en début de processus afin d'obtenir une représentation multimodale.

5.2.1 Le Traitement des Différentes Données dans le modèle d'early fusion

Les caractéristiques audio sont passées à travers des couches de convolution et une couche de pooling adaptatif pour obtenir une représentation dense. Parallèlement, les textes correspondant à chaque segment de parole sont encodés à l'aide d'un tokenizer suivi de CamemBERT, en se concentrant sur le token CLS pour capturer le contexte global de la séquence textuelle. Cette étape assure que les caractéristiques de chaque modalité sont optimisées et prêtes pour une fusion efficace.

5.2.2 La Fusion des Différentes représentations

L'early fusion combine les caractéristiques audio et textuelles à un stade précoce. Les représentations denses des modalités audio et textuelle, obtenues après les étapes de prétraitement, sont concaténées pour former une seule représentation unifiée. Cette représentation combinée passe ensuite dans plusieurs couches fully connected vers la classification. Cette approche de fusion précoce permet au modèle d'apprendre des interactions complexes entre les modalités dès le début du processus de formation, améliorant potentiellement la capacité du modèle à réaliser des prédictions précises sur la base d'informations multimodales intégrées.

6 Résultats

Modèle	Test κ Médiane	Test F1 Médiane
Audio	0.124	35.69%
Texte	0.125	35.75%
Early Fusion	0.127	36.33%
Late Fusion	0.139	35.50%

Table 1: Comparaison des médianes des scores F1 et Kappa de cohen (κ) sur les seeds des données test des différents modèles

Pour assurer la robustesse de nos évaluations, nous avons effectué des estimations sur 10 exécutions différentes avec des seeds différentes, ce qui nous permet d'être plus confiants dans les prédictions de nos modèles. Les valeurs rapportées dans les tableaux sont les médianes des résultats sur ces 10 exécutions (Un notebook contenant ces estimations est disponible pour faciliter la reproductibilité de ces résultats).

Le F1-score est une métrique évaluant la précision et le rappel d'un modèle de classification, un F1-score élevé indique une meilleure performance.

Le Kappa de Cohen est une métrique qui évalue l'accord entre les prédictions d'un modèle et les targets en prenant en compte l'accord qui serait attendu par hasard, donnant une mesure de la performance qui corrige le déséquilibre de classe.

Nous avons opté pour le Kappa de Cohen en métrique principale plutôt que le F1-score car ce dernier peut donner de bons résultats même lorsque la classe minoritaire est mal prédite, ce qui n'est pas forcément adapté à notre projet.

Que ce soit avec le F1-score ou avec le Kappa de Cohen, nos différents modèles ont des performances assez similaires en test, le modèle 'early' étant légèrement plus performant que les autres.

Cependant, étant donné que ces résultats sont assez proches, nous ne pouvons pas conclure. Il nous faudrait pour cela effectuer une étude statistique des résultats, et éventuellement augmenter la taille de l'échantillon des modèles testés.

7 Conclusion

En conclusion, ce projet nous a permis d'explorer en profondeur la prédiction multimodale du tour de parole dans les conversations dyadiques. Malgré les défis rencontrés à causes de ressources limitées en matériel, nous avons réussi à mettre en place des modèles unimodaux et multimodaux pour cette tâche complexe.

La fusion précoce des modalités audio et textuelles a des performances légèrement supérieures à ceux de la fusion tardive, bien que la différence ne semble pas significative. Ces résultats soulignent l'importance de l'intégration de différentes sources d'information pour améliorer la compréhension des conversations. L'ajout de la modalité vidéo aurait certainement permis d'atteindre de meilleurs résultats.

Ce projet nous a permis d'approfondir nos connaissances théoriques et pratiques en deep learning, telles que les avantages et inconvénients des différentes architectures de modèles, ou encore le traitement des différents types de données et leur intégration dans des modèles multimodaux.

La prédiction du tour de parole est un sujet de recherche en deep learning qui n'est pas évident et qui reste à explorer, mais ce projet nous a permis d'acquérir de précieuses compétences en traitement multimodal et en analyse de données, qui seront utiles dans nos futurs travaux de recherche et projets en deep learning.

8 Structure et Documentation du Code

Pour obtenir des informations détaillées sur la structure du code et son organisation, nous vous invitons à visiter le dépôt GitHub de notre projet. Afin de ne pas surcharger le rapport, nous avons décidé de tout mettre ici dans un fichier README.md exhaustif qui fournit toutes les explications nécessaires.

References

[1] Hochreiter, S., and Schmidhuber, J. *Long Short-Term Memory*. Neural Computation, 9(8):1735-1780, 1997.

[2] Martin, L., et al. *CamemBERT: a Tasty French Language Model*. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020.

[3] Devlin, J., et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019.

[4] Muda, L., et al. *Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) techniques*. arXiv preprint arXiv:1003.4083, 2010.

[5] Author, A. *Early, intermediate and late fusion strategies for robust deep learning*. Journal Name, Volume(Issue), Year.

[6] Simonyan, K., and Zisserman, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. arXiv preprint arXiv:1409.1556, 2014.