Détection de bruits parasites dans des enregistrements audios

Maël Alet Julien Alonzo Quentin Saint-Guily

Décembre 2021

Table des matières

1 Introduction

L'objectif de ce document est de présenter les trois différentes méthodes qui sont le plus à même de répondre à la problématique : Détecter des bruits parasites tels que des clics de souris ou des grésillements dans des enregistrements audio. Pour cela, nous avons éliminé les techniques qui se sont prouvées inadaptées et n'avons gardé que celles qui ont montré de bons résultats ou une méthode prometteuse.

Dans un premier temps, le but sera alors de décrire succinctement chacune des méthodes en expliquant brièvement leur possible implémentation. Ensuite nous analyserons et dresserons une critique pour chacune d'entre elles afin de sélectionner celle la plus à même de répondre aux besoins du client.

2 Technique n°1 - Sound Event Detection

2.1 Méthode

Les réseaux de type SED (Sound Event Detection) sont des réseaux de neurones faits pour la détection d'évènements sonores. L'AST (Audio Spectrogram Transformer) est un de ces réseau. Il est pré-entraîné et est capable de détecter des évènements comme des clics de souris et des frappes au clavier.

En pratique, on donne en entrée un enregistrement audio et l'AST nous renvoie en sortie une probabilité pour chaque évènement qu'il est capable de détecter (comme « clic de souris », « frappe au clavier », etc...). Si la probabilité pour un évènement est haute le réseau est plutôt sûr que l'évènement est présent dans l'enregistrement. En revanche, si la probabilité est faible, le réseau est sûr que l'évènement n'y est pas.

Dans notre cas, lorsqu'un un évènement comme « clic » ou « frappe au clavier » sera détecté au-delà d'une certaine probabilité on pourra alors considérer l'enregistrement comme étant défectueux.

2.2 Critiques

— Les SED étant des réseaux de classification, les résultats qu'ils nous renvoient sont facilement analysables car ils ne nécessitent pas de calculs supplémentaires.
— Audio Spectrogram Transformer est un des meilleurs réseau SED en terme de performance. Cependant au vu des tests que nous avons réalisés, nous sommes insatisfait du temps de traitement.
— De manière générale l'utilisation de la mémoire RAM est étrangement très variable. Nos tests ce sont avérés peu concluants sur la nature de ce phénomène qui pénalise fortement cette méthode.
— Après avoir effectué quelques tests sur la détection de clics de souris, nous avons remarqué que la probabilité assignée à la classe « clic » sur des enregistrements avec des clics très facilement audibles est très faible. Au contraire la probabilité assignée à certains enregistrements sans aucun

clic est parfois relativement élevé. De plus, il ne semble pas y avoir assez d'écart entre la probabilité moyenne affectée aux enregistrements sans clic et la probabilité moyenne affectée aux enregistrements avec clics pour permettre de différencier les deux groupes.

3 Technique n°2 - Séparation de sources

3.1 Méthode

Le principe de cette méthode est de séparer le signal source en plusieurs (généralement deux) signaux, un représentant le signal « propre » et un autre le « bruit » pour ensuite les analyser afin de pouvoir caractériser le bruit.

Pour séparer le signal source, nous avons deux méthodes : une méthode basée sur du traitement de signal (Robust PCA) et une autre basée sur le deep-learning avec Asteroid.

3.1.1 Robust Principal Component Analysis (RPCA)

Pour cette méthode, on suppose que le signal source peut-être décomposé en deux parties :

— Une matrice de rang faible pour le bruit, appelée L.

— Une matrice creuse pour la voix, appelée S.

L'objectif du RPCA est donc de transformer le signal source M en deux matrices semblables à celles-ci.

Ainsi cela revient à résoudre le problème d'optimisation suivant :

$$\text{minimize } \|L\|_* + \lambda\|S\|_1$$

$$\text{subject to } L + S = M$$

$$\text{Avec } M, S, L \in \mathbb{R}^{n_1 \times n_2} \text{ et } \lambda = 1/\sqrt{max(n_1, n_2)}$$

De manière générale, la méthode suivante est appliquée :

— D'abord on calcule la Short Time Fourier Transform (STFT) sur le signal source pour obtenir M.

— Ensuite on utilise la RPCA pour obtenir les matrices L et S.

— On utilise alors une technique de Time Frequency Masking pour fabriquer une matrice M_b à partir de L et S et qui servira à générer les matrices X_{voix} et X_{bruit}.

$$\begin{aligned} X_{voie}(m,n) &= M_b(m,n)M(m,n) \\ X_{bruit}(m,n) &= (1 - M_b(m,n))M(m,n) \end{aligned}$$

— Enfin on calcule la Inverse Short Time Fourier Transform (ISTFT) sur chacune des matrices X_{voix} et X_{bruit} afin d'obtenir les deux signaux *voix* et *bruit*.

3.1.2 Asteroid

Asteroid est une boîte à outils pour la séparation de sources de fichiers audio. Ainsi Asteroid offre une multitude de réseaux pré-entraînés à la séparation voix/bruit.

En pratique, on donne en entrée un enregistrement audio et le réseau nous renvoie en sortie plusieurs (généralement deux) signaux.

3.2 Critiques

3.2.1 RPCA

— Le principal problème de cette méthode est la difficulté de trouver la bonne formule pour le Time Frequency Masking. La totalité des masques que nous avons testé nous a retourner des résultats moyennement satisfaisant voire inutilisable. Ces résultats nous ont surtout appris que trouver un masque optimal relève d'un vrai problème mathématique qui nous prendrait trop de temps à résoudre.

— Le calcul de la RPCA même utilisé pour obtenir les matrices L et S peut prendre plusieurs minutes dans certains cas, notamment si le signal source est difficilement séparable. Un tel temps de calcul n'est pas acceptable, surtout que l'intégralité des fichiers ne comportant pas de défauts seront par nature difficilement séparables et donc longs à traiter.

3.2.2 Asteroid

— L'utilisation d'Asteroid nous permet de résoudre le problème du calcul du masque grâce à l'utilisation de réseau de neurones déjà entraîné. Ainsi les résultats obtenus avec cette méthode sont encourageants.

— De plus, le temps de calcul ne dure que quelques secondes pour des fichiers audio de durée relativement courte.

— L'irrégularité sur le nombre de signaux en sortie nous pose léger problème mais il reste possible d'analyser ces signaux de manière plus générique. De manière générale, il nous reste encore à trouver la meilleure solution pour analyser les différents signaux en sortie.

4 Technique n°3 - Multiclass Audio Segmentation

4.1 Méthode

Les réseaux de type « Multiclass Audio Segmentation » sont un type de réseau de neurones qui ont pour rôle de distinguer, dans un fichier audio, la nature des activités sonores (par exemple, parole, moteur, oiseau, bris de verre, etc...). Il s'agit alors de donner à chaque partie du signal une étiquette qui décrit cette partie.

En pratique, on donne en entrée un enregistrement audio et le réseau nous renvoie en sortie une liste des différentes étiquettes pour chacune des parties de

l'enregistrement. Il ne nous reste alors qu'à vérifier que cette liste ne contienne pas les étiquettes « clic » , « clavier » , etc. . .

4.2 Critiques

— Au vue des résultats annoncés, cette méthode possède un temps d'exécution relativement court tout en ayant une précision globale très satisfaisante.
— Nous n'avons pas trouvé de réseau déjà existant qui serait adapté à notre problème. Il faudrait donc le concevoir et l'entraîner nous-même, cependant nous n'avons ni le temps, ni les compétences.

5 Conclusion

La méthode 1, Sound Event Detection, n'a pas été retenue car elle manque de précision et a une trop grande durée d'exécution.

La troisième méthode est également mise de côté, car aucun réseau n'est disponible. Créer le notre serait trop compliqué et trop long à notre niveau.

Il ne reste alors que la méthode 2. Nous n'avons pas gardé la RPCA, car il est fastidieux de trouver un masque. De plus, les temps de calcul prennent parfois trop de temps.

Finalement, nous avons sélectionné la méthode utilisant Asteroid. Nous avons obtenu les meilleurs résultats grâce à elle, pour un temps de calcul satisfaisant.