Présentation Finale

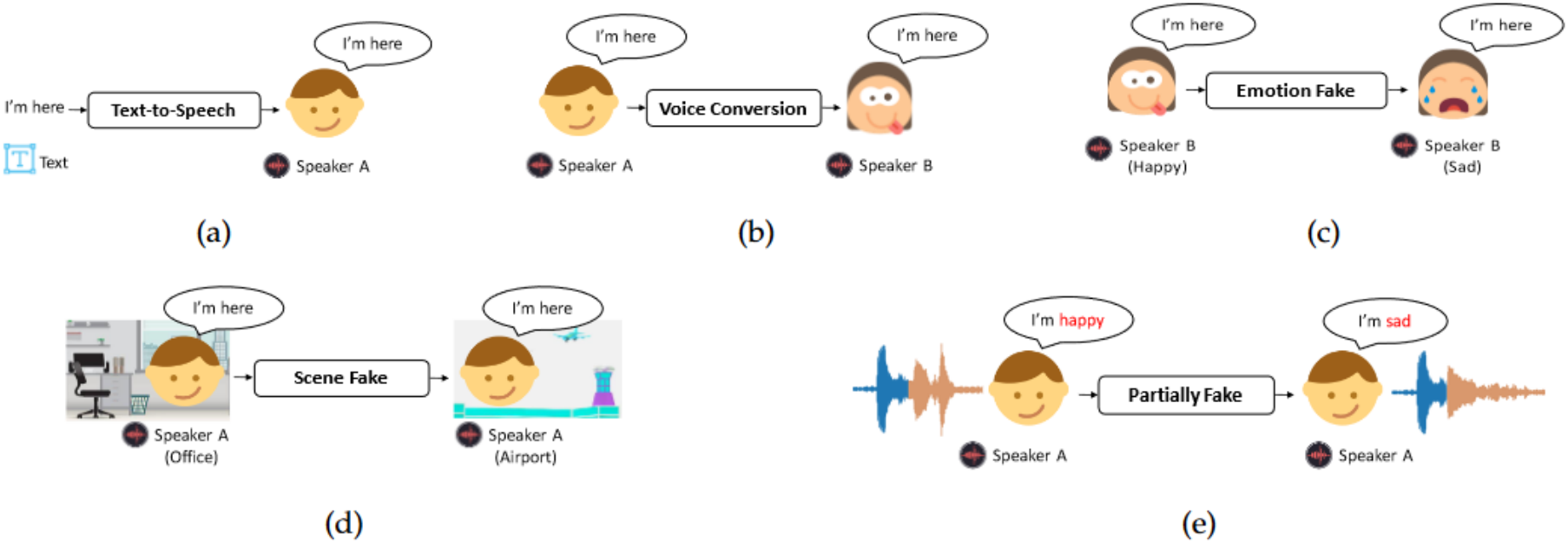
Amélioration des systèmes de vérification du locuteur contre les attaques Deepfakes

Encadré par **Réda Dehak** et **Théo Lepage**

Contexte

Usurpation d'identité avec de l'audio deepfake

Différents types de deepfakes [10]:



Objectif du système:

- Détecter les *artefacts* laissé par l'algorithme dans l'audio

Problèmes:

- Type d'attaque inconnu
- Compression de l'audio sur les plateformes en lignes
- Les attaquants ont accès aux systèmes de détections pour améliorer la qualité de leur algorithme

Schéma du système proposé par l'atelier ASVspoof 2019 [9]

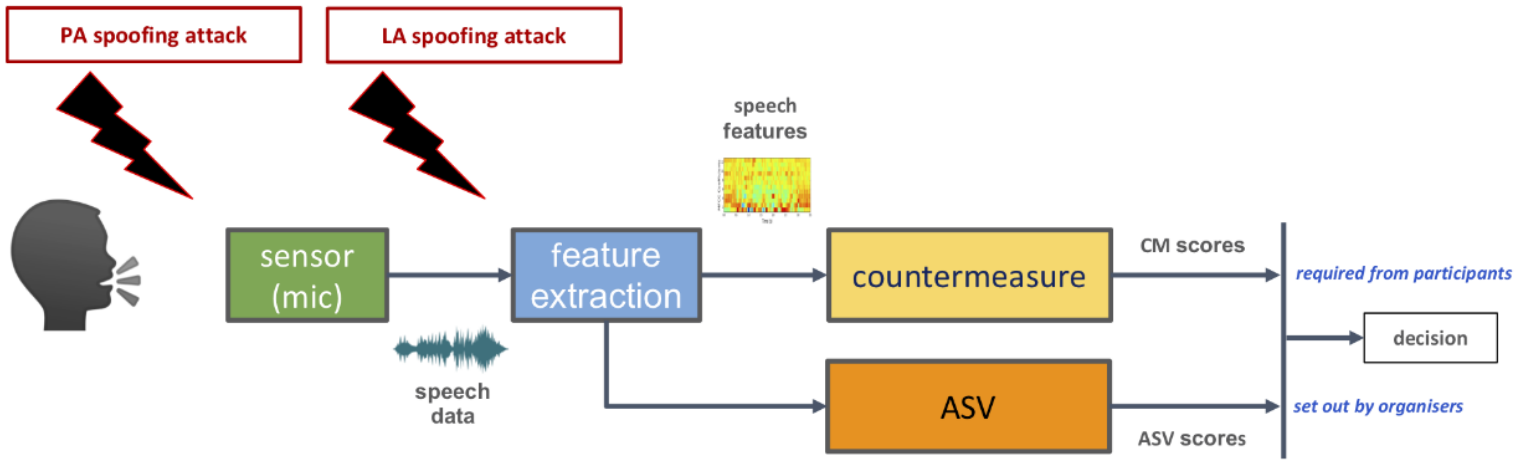


Figure 1: LA access and PA access adopted for ASVspoof 2019

Etat de l'art

Deux systèmes étudiés:

- RawNet2 [7]
- AASIST [3]

Proposition:

Améliorer les performances du système en intégrant un modèle auto-supervisé pré-entraîné comme encodeur [6] [4]

Etat de l'art

RawNet2

- a introduit le fait d'utiliser l'audio brut en entrée d'un système de détection
- modèle compact ($\approx 25 * 10^6$ paramètres)
- crée pour le challenge ASVspoof 2019 et utilisé comme *baseline* dans l'édition 2021

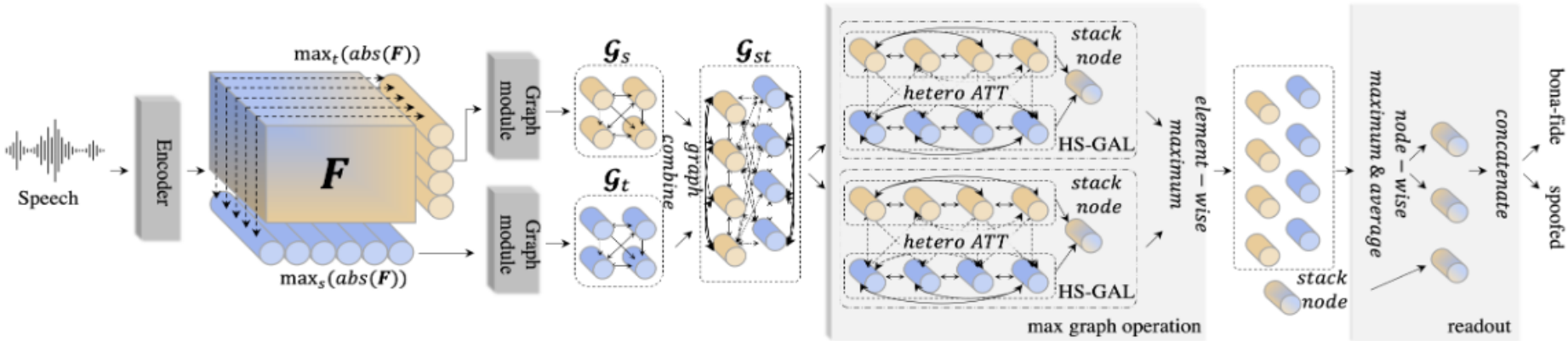
Layer	Input:**64000** samples	Output shape
Fixed Sinc filters	Conv(**129**,1,128) Maxpooling(3) BN & LeakyReLU	(**21290**,128)
Res block	{ BN & LeakyReLU Conv(3,1,128) BN & LeakyReLU Conv(3,1,128) Maxpooling(3) FMS } × 2	(**2365**,128)
Res block	{ BN & LeakyReLU Conv(3,1, **512**) BN & LeakyReLU Conv(3,1, **512**) Maxpooling(3) FMS } × 4	(**29**,512)
GRU	GRU(1024)	(1024)
FC	1024	(1024)
Output	**1024**	**2**

Architecture du modèle RawNet2 [7]

Etat de l'art

AASIST

- se base sur un encodeur similaire au RawNet2
- sépare la dimension fréquentielle et la dimension temporelle à la sortie de l'encodeur
- utilise des GANs (Graph Attention Networks)



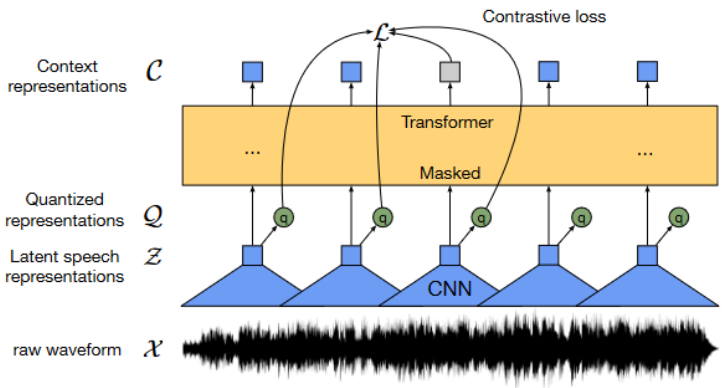
Architecture du modèle AASIST [3]

Proposition

Extension du RawNet2 avec le modèle Wav2Vec2.0

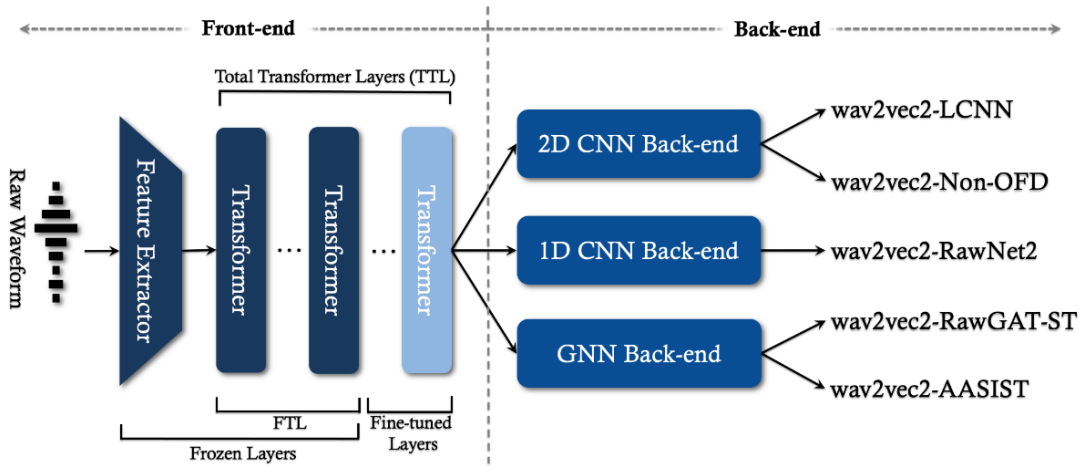
Modèle Wav2Vec2.0[1]:

- Modèle auto-supervisé pré-entraîné sur de très gros corpus de voix
- S'entraîne en essayant de reconstruire des parties de l'audio qui ont été masqué



Introduction de deux nouveaux paramètres[4]: **TTL** et **FTL**

- TTL: # de couches Transformers sélectionnées
- FTL: # de couches Transformers fixées



Proposition

Adaptation du modèle MHFA pour la détection de deepfake

Modèle utilisé originellement pour de la vérification du locuteur [6] [5]

- *Fine-tune* un modèle auto-supervisé pré-entraîné en *front-end*
- Effectue un calcul d'attention entre les sorties des différentes couches Transformers
- Applique une régularisation des poids sur l'encodeur pour ne pas trop s'éloigner des paramètres de base

Structure du modèle[6]:

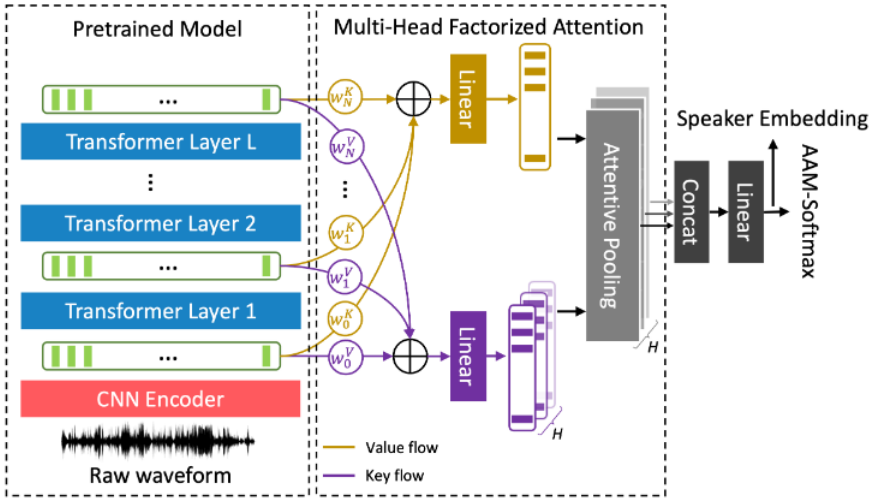


Fig. 3. (c) Proposed Multi-Head Factorized Attentive Pooling.

Pour adapter ce modèle à une tâche de détection:

- Ajout d'une couche linéaire à la fin du réseau avec deux éléments en sortie

Protocole d'évaluation

Jeu de donnée *ASVspoff 2019 LA*

Les jeux de données utilisés sont issus de l'ensemble ***ASVspoof2019 LA***:

Jeu de donnée	**# d'échantillons**	**% d'échantillons factices**
Entraînement	25 380	89.83%
Développement	24 844	89.74%
Évaluation	71 237	89.67%

Equal Error Rate (EER) sur la tâche de détection: Probabilité d'erreur lorsque l'on choisis un seuil pour lequel on a autant de faux positifs que de faux négatifs sur le jeu de données d'évaluation.

min-DCF: Score qui prend en compte la probabilité de la catégorie cible (P_{tar}), et qui attribue un coût aux deux types d'erreurs: faux positif (C_{FA}) et faux négatif (C_{miss}).

$$C_{det}(P_{miss}, P_{FA}) = C_{miss} P_{miss} P_{tar} + C_{FA} P_{FA} (1 - P_{tar})$$

Protocole d'évaluation

Métriques d'évaluation

min-tDCF: Similaire au min-DCF, mais prend en compte les scores d'un système de vérification du locuteur (ASV) en plus des scores d'un système de détection (CM). Cela introduit 4 types d'erreurs:

- le système ASV rejette un positif (miss):

$$P_a(s,t) = (1 - P^{cm}_{miss}(s))xP^{asv}_{miss}(t)$$

- le système ASV accepte un négatif (False-Alarm):

$$P_B(s,t) = (1 - P^{cm}_{miss}(s))xP^{asv}_{FA}(t)$$

- le système CM accepte un deepfake:

$$P_c(s,t) = P^{cm}_{FA}(s)x(1 - P^{asv}_{miss}(t))$$

- le système CM rejette un positif:

$$P_d(s) = P^{cm}_{miss}(s)$$

Il faut aussi définir des coûts d'erreurs associés: $C^{asv}_{miss}, C^{asv}_{FA}, C^{cm}_{miss}, C^{cm}_{FA}$

$$\text{t-}DCF(s,t) = C^{asv}_{miss}P_{tar}P_a(s,t) + C^{asv}_{FA}P_{non}P_b(s,t) + C^{cm}_{FA}P_{spoof}P_c(s,t) + C^{cm}_{miss}P_{tar}P_d(s)$$

Jeu de données d'évaluation: ***ASVspoof2019 LA evaluation***

Modèle	**EER**	**min-tDCF**
RawNet2	5.67%	0.1306
AASIST	**1.47%**	**0.0481**
Wav2Vec2.0 + RawNet2	**0.204%**	**0.0068**
MHFA	4.22%	0.1074

Challenge ASVSPOOF5

Spécifications

Trois nouveaux jeux de données:

Jeu de donnée	**# d'échantillons**	**% d'échantillons factices**
Entraînement	182 357	89.69%
Développement	140 950	60.11%
Évaluation	40 765	???

Deux conditions:

- Fermée: seule les données d'*entraînements* fournies peuvent être utilisées
- Ouverte: même condition, mais il est possible d'utiliser des modèles auto-supervisés, pré-entraînés sur d'autres jeux de données

Quelques dates:

- Début du challenge: 20 Mai 2024
- Soumission des scores: 23 Juillet 2024
- Soumission de l'article: 31 Juillet 2024
- Atelier ASVspoof 2024 à Interspeech: 31 Août 2024

Protocole d'évaluation

Configuration ASVSPOOF5

Le jeu de données de *développement* a été divisé en 2 parties pour former un jeu de validation et un jeu d'évaluation

Jeu de donnée	**# d'échantillons**	**% d'échantillons factices**
Validation	37 091	77.77%
Évaluation	103 859	77.77%

Les systèmes sont entraînés sur l'ensemble du jeu de données d'*entraînement* et les meilleurs modèles sont sélectionnés à partir des résultats sur l'ensemble de *validation*.

Augmentation des données

Augmentation utilisée en vérification du locuteur (dénoté *basique*):

- Réverbération
- Ajout d'audio par dessus l'extrait: bruit, musique, voix

Augmentation spécifique à la détection de deepfake:

- *RawBoost*[8]: Applique un effet de *boost* ou de distorsion au signal, selon différents algorithmes
- *Codecs*: Applique une compression puis décompression (avec perte) selon un certain codec

Résultats actuels du challenge

Pour la condition fermée:

Modèle	**Augmentation**	**min-DCF**	**EER**
ResNet	non	0.4026	22%
ResNet	codec	0.4472	23%
ResNet	***basique + codec***	***0.4466***	***21%***
AASIST	basique	0.5497	25%
RW-ResNet	non	0.7930	40%
RW-ResNet	codec	0.7677	40%
MFCC-RW-Resnet	non	0.4857	24%
MFCC-RW-Resnet	codec	0.5081	24%

Résultats actuels du challenge

Pour la condition ouverte:

Modèle	**Augmentation**	**min-DCF**	**EER**
MHFA (fixée)	non	0.2073	8.36%
MHFA (fixée)	basique	0.1365	5.66%
MHFA	non	0.1049	4.78%
MHFA	***basique***	***0.0804***	***3.15%***
MHFA	RawBoost [8]	0.8363	32%
MHFA	RawBoost [8]	0.7160	28%

Conclusions et perspectives

- Les données d'entraînements et l'augmentation joue un rôle primordiale dans la performance d'un système de détection
- Les encodeurs auto-supervisés (notamment le modèle WavLM [2]) améliorent les performances des systèmes de détection
- Exploiter la fusion de différents modèles pourrait améliorer les performances

Bibliographie I

[1] Alexei Baevski et al. *wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations*. 2020. arXiv: 2006.11477 [cs.CL].

[2] Sanyuan Chen et al. "WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing". In: *IEEE Journal of Selected Topics in Signal Processing* 16.6 (Oct. 2022), pp. 1505–1518. ISSN: 1941-0484. DOI: 10.1109/jstsp.2022.3188113. URL: http://dx.doi.org/10.1109/JSTSP.2022.3188113.

[3] Jee-weon Jung et al. *AASIST: Audio Anti-Spoofing using Integrated Spectro-Temporal Graph Attention Networks*. 2021. arXiv: 2110.01200 [eess.AS].

Bibliographie II

[4] Taein Kang et al. *Experimental Study: Enhancing Voice Spoofing Detection Models with wav2vec 2.0*. 2024. arXiv: 2402.17127 [cs.SD].

[5] Victor Miara, Theo Lepage, and Reda Dehak. *Towards Supervised Performance on Speaker Verification with Self-Supervised Learning by Leveraging Large-Scale ASR Models*. 2024. arXiv: 2406.02285 [eess.AS]. URL: https://arxiv.org/abs/2406.02285.

[6] Junyi Peng et al. *An attention-based backend allowing efficient fine-tuning of transformer models for speaker verification*. 2022. arXiv: 2210.01273 [eess.AS]. URL: https://arxiv.org/abs/2210.01273.

[7] Hemlata Tak et al. *End-to-end anti-spoofing with RawNet2*. 2021. arXiv: 2011.01108 [eess.AS].

Bibliographie III

[8] Hemlata Tak et al. *RawBoost: A Raw Data Boosting and Augmentation Method applied to Automatic Speaker Verification Anti-Spoofing*. 2022. arXiv: 2111.04433 [eess.AS]. URL: https://arxiv.org/abs/2111.04433.

[9] Xin Wang et al. *ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech*. 2020. arXiv: 1911.01601 [eess.AS].

[10] Jiangyan Yi et al. *Audio Deepfake Detection: A Survey*. 2023. arXiv: 2308.14970 [cs.SD]. URL: https://arxiv.org/abs/2308.14970.

Questions

Merci de votre attention !

Des questions ?