

Projet classification des émotions

MAIZA Hichem

M2 ATAL

1 Introduction

Ce projet présente la mise en application du module traitement de la parole. L'objectif de ce projet est la classification des émotions à partir des signaux générés d'une façon spontanée et dans des conditions réelles, sans avoir recours à des techniques de traitement du langage naturel. La particularité de ce projet se manifeste dans l'utilisation des signaux pour détecter les émotions.

2 Système Proposé

Dans cette partie du rapport, je décris brièvement le système de l'état de l'art proposé par la communauté en traitement de la parole dans le cadre de la détection d'émotions. Le système comporte cinq étapes nécessaires pour une bonne démarche et une structuration logique du projet:

1. Sélection des personnes
2. Acquisition des signaux
3. Traitement des signaux
4. Extraction des descripteurs
5. Classification

Les deux premières étapes consistent à préparer le corpus, sélectionner les personnes qui vont représenter les données d'apprentissage et de test. Le corpus utilisé dans le cadre de ce projet est le corpus AIBO qui comporte les données audio de chaque enfant, son sexe et son âge. Ces données ont été étiquetées par des linguistes de façon à avoir une étiquette pour chaque mot et un seul label pour tout le segment audio. La troisième étape consiste à pré traiter le signal audio et le filtrer. Ensuite, l'extraction des descripteurs qui sont représentés par des vecteurs MFCCs dans le cadre de ce projet. J'ai ajouté d'autres descripteurs comme la moyenne, le 'standard deviation', le 'kurtosis' et le 'skewness'. La dernière étape consiste à appliquer les différentes techniques de classification. Dans le cadre de ce projet on ne s'intéresse que dans les deux dernières étapes.

2.1 Descripteurs

Les descripteurs MFCC sont générés à l'aide de l'outil openSmile et se sont résumés dans un tableau comme montre la Figure 1

	MFCC	classe_2c	classe_5c	file_name	kurtosis	mean	skewness	std	value_2c	value_5c
0	[0.005667331, 0.0003117731, 0.003355558, 53.0...	IDL	N	Ohm_01_015_00	357.183887	22.074596	18.693648	185.900529	1	1
1	[0.01041963, 0.0002702086, 0.01014942, 61.0...3...	IDL	N	Ohm_01_016_00	370.521908	31.192448	19.195588	324.308852	1	1
2	[0.01909119, 0.0002510243, 0.01884016, 60.0...1...	IDL	N	Ohm_01_017_00	353.025960	34.716302	18.528228	271.927149	1	1
3	[0.01374097, 0.0002331673, 0.0135078, 66.0...20...	IDL	N	Ohm_01_018_00	351.132968	23.971229	18.470221	191.195626	0.9	0.9
4	[0.001530198, 0.0002304439, 0.001299754, 49.0....	IDL	N	Ohm_01_018_01	367.001161	34.619512	19.073093	394.509343	1	1

Figure 1: Descripteurs

3 Approche

3.1 Approche Basique

L'approche basique consiste à entraîner des classifieurs de l'état de l'art avec des vecteurs MFCC. J'ai entraîné plusieurs classifieurs dont : SVM, naïve bayes, GMM, Logistic regression (2 classes).

3.1.1 GMM-Classifier

Les premières essais ont été faites en entraînant des gmms pour chaque classe, comme présenté dans l'article. Les résultats sont résumés dans le tableau de la Figure 2. Les résultats trouvés sont proches des celles présentés dans l'article mais ils sont moins bons et c'est dû probablement aux données de test qu'ils ont utilisées. dans mon cas j'ai utilisés toutes les données AIBO-M pour le test comme décrit dans l'article alors que j'ai tous les labels et je pourrais utiliser une partie de test, je suppose donc que les participants ont utilisé une partie de corpus AIBO-M pour l'entraînement. Le résultats sont présentés dans la Figure 2.

3.1.2 SVM-Classifier

La deuxième partie consiste à entraîner un SVM. Le but dans ce cas-là est d'améliorer les résultats obtenus par le système de classification en utilisant des gmm. Le tableau 2 de la Figure 3 présente une comparaison entre un SVM et une régression logistique qui peut être appliquée dans le cas de deux classes. la précision a augmenté de 0.1 et le rappel restent presque le même.

	2 classes	5 classes
précision	0.555	0.414
rappel	0.631	0.360

Figure 2: Classification par GMM

	2 classes	5 classes		2 classes
précision	0.431	0.314	précision	0.537
rappel	0.656	0.560	rappel	0.648

Figure 3: Classification par SVM

3.2 Approche Neuronale

3.2.1 Introduction

Cette approche consiste à utiliser directement le signal brut sans passer par l'extraction des descripteurs de type MFCC par exemple. L'objectif de cette approche est d'entraîner directement le signal brut sur un réseau de neurones d'architecture particulière qui devrait ensuite trouver la représentation des descripteurs qui pourrait décrire le signal d'entrée ils pourraient aussi rassembler à des vecteurs MFCC ou d'autres.

3.2.2 Données et Difficultés

Les fichiers audio qui constituent le corpus AIBO n'ont pas les mêmes longueurs. Ainsi, chaque fichier audio a son propre vecteur. Deux solutions qui m'ont semblées applicables pour résoudre ce problème. La première est de chercher le signal le plus long et le considérer comme référence c'est à dire de remplir tous les autres vecteurs avec des valeurs très faibles de l'ordre de 0.001 de telle sorte qu'ils auront tous les mêmes longueurs. La deuxième solution est de passer par une couche particulière qui permet de modéliser les signaux mais comme on ne peut pas changer la topologie du réseau en cours d'exécution. Ce point n'est pas donné dans l'article

3.2.3 Implémentations

La lecture des signaux bruts a été faite en utilisant le module wavefile, la génération des fichiers nécessite plus d'une heure sur une machine de processeur i5 et aussi leurs prétraitements. l'implémentation a été faite à l'aide de l'outil Keras. En se basant sur la même architecture proposée par le deuxième article. J'ai bien décrit le système dans le notebook, reste à l'entraîner sur des machines avec des GPU (Rq: Keras n'est pas installé sur les machines du Mans).