A speech emotion recognition dataset for the English language

Fatemeh Ahmadvand
Politecnico di Torino
s301384
Torino, Italy
s301384@studenti.polito.it

Iván Contreras
Politecnico di Torino
s301962
Torino, Italy
s301962@studenti.polito.it

***Abstract*—Emotions are complex multidimensional concepts, in this report, we have built a model to classify the emotion of that given a short snippet of audio containing a spoken sentence in English predicts whether the emotion of the voice is happiness, sadness, anger, fear, disgusted, surprised, and neutral. This study aims to create and evaluate two classifiers, one SVM classifier and one linear Neural Network to predict the sentiment of audio using the extracted features. The proposed system is tested on a dataset of almost 10k samples and is able to classify up to 62.7% of voices accurately(using F1 Score as evaluation method).**

I. PROBLEM OVERVIEW

Speech emotion recognition is a classification problem where an input sample (audio) needs to be classified into a few predefined emotions. Of course, the challenge in this problem goes beyond technical. The objective of the project is to predict the emotion of the audio sentence [1]. The dataset provided consists of two different parts:

- A development set of 9597 objects is characterized by the emotion column that represents the emotion expressed in the audio sentence and the column of the filename containing the name of the audio file. This development set does not contain duplicate objects.

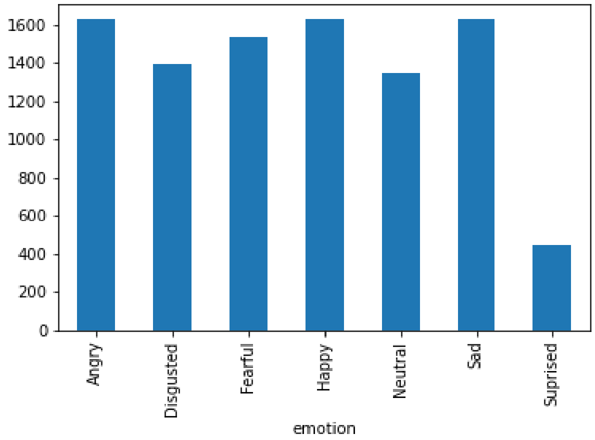

Fig.1: Count of Emotions

The development dataset with 1625 recordings for angry, happy, and sad emotions, 1535 recordings for the fearful emotion, 1397 recordings for the disgusted emotion, 1346 recordings for the neutral emotion, and 444 recordings for the surprised emotion, as depicted in Fig. 1. The duration of each of the audio recordings is from 2 to 3 s [2].

- An evaluation set of 3201 data points with the same structure as the development set, except for the emotion column.

II. PROPOSED APPROACH

A. Data exploration

The audio files are recorded in 32-bit float WAV format [2]. We loaded audio files and used their natural sample rate since all the files are mono-channel and have the same sample rate. A waveform plot for every emotion of a randomly selected sample from the Development dataset is presented in Fig. 2.

Here, the X-axis is representing different time frames measured in seconds, and the Y-axis represents amplitude that indicates the amount of air compression ($> zero$) or rarefaction ($< zero$) induced by a moving object, such as the vocal cords, and pressure equilibrium point (= zero) denotes silence. The typical range of the Y-axis is [1, -1]. However, we did not perform that scaling here so that we can visualize the waveforms clearly with auto-scaling of the python librosa library [2]. Using the same scale for all the samples with different amplitudes will not be visually comparable which is why auto-scale is used for visualization purposes only. The features generated from the associated audio-recording of each waveform can be eventually scaled to a normalized value to train a machine learning-driven speech emotion recognition (SER) model [2].

As can be seen in Fig. 3, we explored tags distribution for the amount of data for each tag to avoid overfitting some categories and underfitting others. Then, the length of the audio files depending on the tag is explored as well to check if the audio length is correlated to the emotion tag; the result of this analysis is that the distribution of the tags can be a characteristic of the population where the audios are coming from, therefore balancing the data can cause a shift between the training and production data.

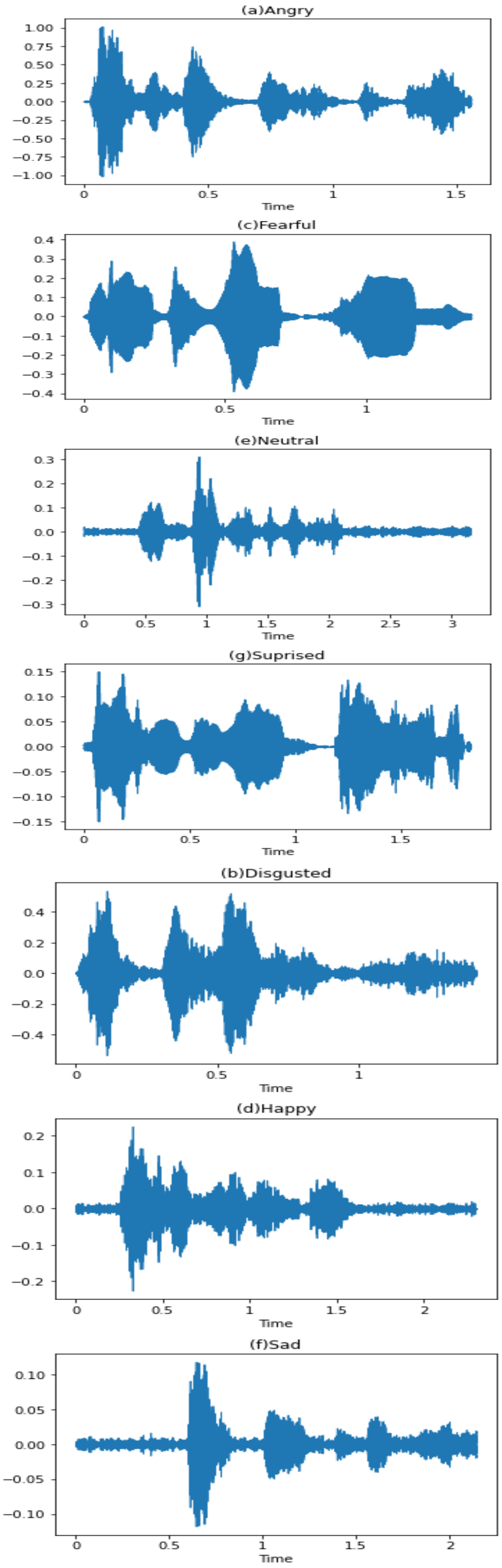

Fig.2:Sample waveform plot of a randomly selected sample of every emotional state, (a) happiness, (b) sadness, (c) anger, (d) fear, (e) disgust, (f)surprise ,and (g)neutral

emotions of the development dataset

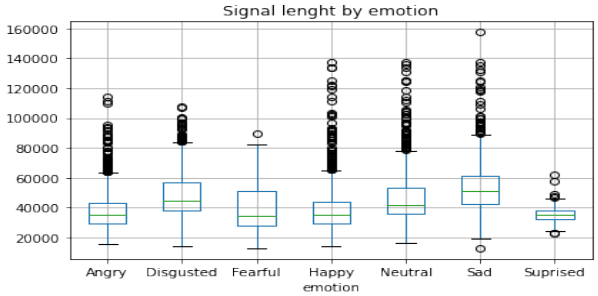

Fig.3: Exploring length distribution

B. Preprocessing

The original speech after preprocessing can better meet the practical needs, and the accuracy of extracted features is also higher. Speech signal preprocessing is of great significance in the process of speech emotion analysis. There are many aspects of speech signal preprocessing, but generally speaking, it is no more than the main steps: sampling and quantization, pre-emphasis, and frame windowing (Priya et al., 2020; Wang, 2019). We used .WAV format to remove any external noises. Each data file is then given a unique filename.

- Trimming silence: Silence has no relevant information, so it is removed from the start and the end of every audio file, as shown in Fig. 4.

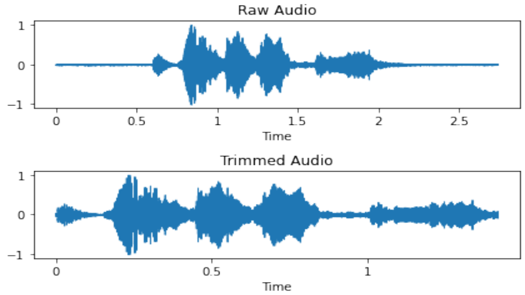

Fig.4:Trimming silence audio

- Adding length as a feature: To process the audio, we needed to resample every audio signal to make them equally sized, therefore the original length information will be lost. The length of the audio is correlated to the emotion (found in the exploratory phase Fig. 5.) and should be kept, that is the reason it is added as a feature, .

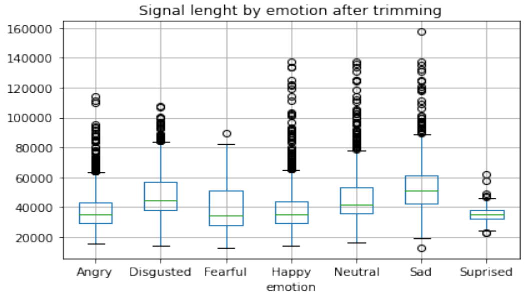

Fig.5:Exploring length distribution after trimming

- Turning into spectrogram: After splitting and padding the audio files, we used the Librosa library to convert the audio to Mel scale spectrograms. These are visual representations of the spectrum of frequencies of a signal over time that are widely used for audio classification. It encapsulates the main 3 dimensions of the audio signal (time, amplitude, and frequency), as shown in Fig. 6. To process the data every audio signal should have the same size.
- Managing size difference: All the audio signals need to be the same size to be processed, in order to accomplish this the hop_length to build each spectrogram is variable depending on the signal size as shown in Eq. 1, this way larger audio signals use smaller hop length.

$$hop\_length = \frac{signal\_length}{spectrogram\_width}$$

- Averaging frequencies (dimension reduction): To make the model efficient because of the restriction of the resources for this project, the spectrogram is reduced from a matrix to a vector containing the average of the frequencies.
- Scaling the vectors: The content of the vectors is scaled using a standard scaler (z score) to guarantee better processing.

C. Model selection

Having prepared our data, we have to select one model and tune and validate it among the following models:

- One-Dimensional Network (1D-NN): The neural network is well-suited for this problem because of the number of dimensions and the non-linear nature of the data. We selected a one-dimensional network over a convolutional network due to the limited processing power we had to finish this project.
- Support vector machines (SVMs) are powerful yet flexible supervised machine learning methods used for classification, regression, and, outliers' detection. SVMs are very efficient in high-dimensional spaces and generally are used in classification problems. SVMs are popular and memory-efficient because they use a subset of training points in the decision function. We selected Support Vector Machine due to its ability to high dimensional data, we tested 2 kernels by using a grid search polynomial and RBF. we tested each kernel with different regularization values to avoid overfitting and evaluated its performance by using the F1 score.

After running two models, the SVM model performs better in comparison to 1D-NN Table I.

Models	macro avg
SVM	62.7
1D-NN	0.06

TABLE I
COMPARISON OF MODELS

D. Hyperparameters tuning

Regarding the SVM model, the following hyperparameters are adjusted:

- Kernel: The main hyperparameter of the SVM is the kernel. It maps the observations into some feature space. The default value of the kernel would be 'rbf'. It represents the degree of the 'poly' kernel function and will be ignored by all other kernels.[6]
- C: inverse of regularization strength; must be a positive float. Like in support vector machines, smaller values specify stronger regularization.[7]

We performed an 80/20 Train-Test split then used the GridSearchCV() method and passed the parameters and estimator or model into as the arguments to get the best parameters for our model. After that, we fit it on the train features and train labels. We found the best parameters by using the cv.best_params_. In Table II it is possible to see parameter values and Every model present in Table 2 is tested, we tried every combination of hyperparameters related to the respective model and achieved the best F1 score obtained on Public Leaderboard and during local tests. (Table II)

Models	Hyperparameters	Values
SVM	kernels	'rbf', 'poly'
	C	0.001, 0.01, 1.0, 5.0, 10 ,25
1D-NN	Epochs	0...60

TABLE II
COMPARISON OF MODELS

III. RESULTS

We can clearly see the GridSearchCV has shown the best parameter for the model. It is 25 for C and 'rbf' kernel. We Used it for improving the accuracy of the model in the given dataset. This could create perfect or almost perfect predictions for those records. However this is only a supposition, and the real cause of the higher values in the public setting could be others. Although it should also be noted that the SVM Model is the fastest to train and predict, it also achieves 62.7% accuracy while classifying.

IV. DISCUSSION

The English speech emotion recognition method based on speech recognition designed in this paper can recognize the emotion of English speech very well, but there are still shortcomings and defects in this topic, which need to be further improved: the performance of the neural network was far from the desired (Fig 7.), the cause has not been investigated yet, however the most probable reason is an excess of droputs that underfit the data.

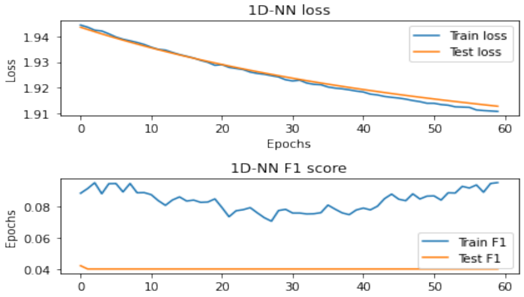

Fig.7: Training and test loss and F1 score for the 1D-NN.

SVM is perfect to face this problem, however in the case that more samples are added and the retraining process scales to tenths of thousands of samples, the training load will be similar to a CNN.

REFERENCES

[1] ”Speech Emotion Recognition Project Using Machine Learning” https://www.projectpro.io/article/speech-emotion-recognition-project-using-machine-learning/573

[2] BanglaSER: A speech emotion recognition dataset for the Bangla language, Rakesh KumarDas, NahidulIslam, Md. RayhanAhmed, SalekulIslam, SwakkharShatabda, A.K.M. MuzahidulIslam (2022) . Journal of Data in Brief, Volume 42, June 2022, 108091, https://doi.org/10.1016/j.dib.2022.108091

[3] Liu, M. English speech emotion recognition method based on speech recognition. International Journal of Speech Technology (2022). https://doi.org/10.1007/s10772-021-09955-4

[4] Priya, R. V., Vijayakumar, V., Ta, V. J. (2020). MQSMER: A mixed quadratic shape model with optimal fuzzy membership functions for emotion recognition. Neural Computing and Applications, 32(8), 3165–3182.

[5] Wang, Y. (2019). The function development of network teaching system to English pronunciation and tone in the background of the internet of things. Journal of Intelligent and Fuzzy Systems, 37(5), 5965–5972.

[6] https://philipppro.github.io/Hyperparameters_svm

[7] https://scikit-learn.org

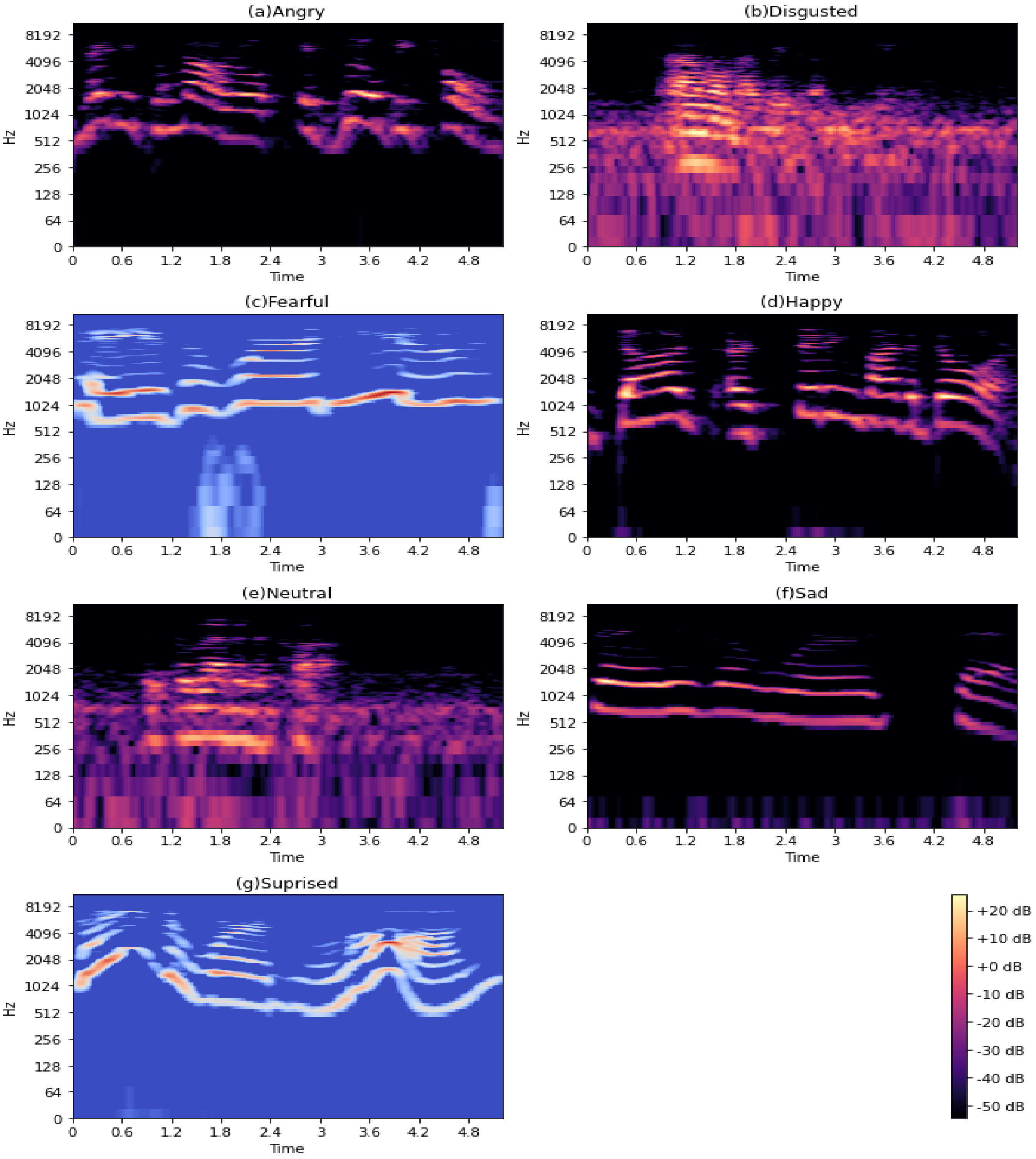

Fig.6: Spectrograms of every emotional state, (a) happiness, (b) sadness, (c) anger, (d) fear, (e) disgust, (f)surprise ,and (g)neutral emotions of the development dataset