EMOVO Dataset:
Exploiting CNNs for Speech Emotion Recognition

Sofia Cazzaniga and Davide Mor

University of Milano-Bicocca

Abstract. The Speech Emotion Recognition (SER) is a technology based on identifying the human emotions by using the spoken language. SER is focused in understanding emotions like: happiness, sadness, fear, anger and other ones, by relying on the characteristics of audio signals. A popular approach to address SER is the utilization of Convolutional Neural Networks (CNNs).

The use of CNNs in SER offer significant advantages, as these networks can automatically learn relevant features from audio data and recognize complex patterns within the signals. Through convolution, CNNs can capture spatial and temporal information in audio signals, adding more information to the acoustic features associated with emotions.

Keywords: SER · Classification · CNN · Transfer learning · Data augmentation

1 Introduction

The project proposed consists in performing speech emotion recognition on the EMOVO dataset, by using a deep neural network. In particular, to mitigate the data scarcity problem brought by the small dataset, two approaches are explored: transfer learning and data augmentation.

Transfer learning is a machine learning technique that exploits the knowledge previously learned by a model and use it as a starting point for a new task or for the same task but in a different domain. For this project, two more datasets are used for that purpose: RAVDESS and CREMA-D. In this case the task is the same (speech emotion recognition), but it is applied on two different domains since Emovo is an Italian dataset while the other two are english datasets.

Data augmentation, instead, consists in synthetically creating new data from the existing one, using some types of transformations applied to the audio signals of the EMOVO dataset.

2 Material and Methods

2.1 Datasets

In the project three datasets are used: EMOVO, RAVDESS and CREMA-D.

EMOVO is an Italian dataset composed of audios from 6 actors (3 males and 3 females), playing 14 sentences while simulating 6 emotional states: disgust, fear, anger, joy, surprise and sad. The sentences are composed by short phrases, long phrases, questions and phrases “nonsense”.

RAVDESS is an American dataset containing audio-visual recordings from 24 actors (12 female, 12 male) playing 2 sentences in 7 emotions (calm, happy, sad, angry, fear, surprise, disgust).

CREMA-D is dataset containing audio-visual recordings from 91 actors (48 male and 43 female), performing 12 sentences using 6 different emotions (anger, disgust, fear, happy, neutral and sad) and four different emotion levels (low, medium, high and unspecified).

For the last two datasets only the audio signals are used for the purpose of this project.

2.2 Pipeline

2.2.1 Feature extraction and segmentation

The first step of the project consists in extracting some characteristic features from the audio signals of the three datasets:

- Mel-spectrograms: They are visual representations of sound signals in the frequency domain. They are obtained by calculating the Short-Time Fourier Transform (STFT) of an audio signal and then mapping the resulting frequencies to the Mel scale, which approximates the human auditory system’s perception of pitch.

- Mel Frequency Cepstral Coefficients (MFCCs): They are derived from the Mel spectrogram by applying a logarithmic compression and then calculating the discrete cosine transform (DCT) on the Mel filterbank energies.

- Chroma STFT: Chroma STFT is a representation of audio signals in the pitch class space, capturing the twelve different musical pitch classes (corresponding to notes in the Western musical scale). It is obtained by summing the STFT magnitude spectrum’s energy within each pitch class. They help in identifying the harmonic content of audio signals and their tonal characteristics.

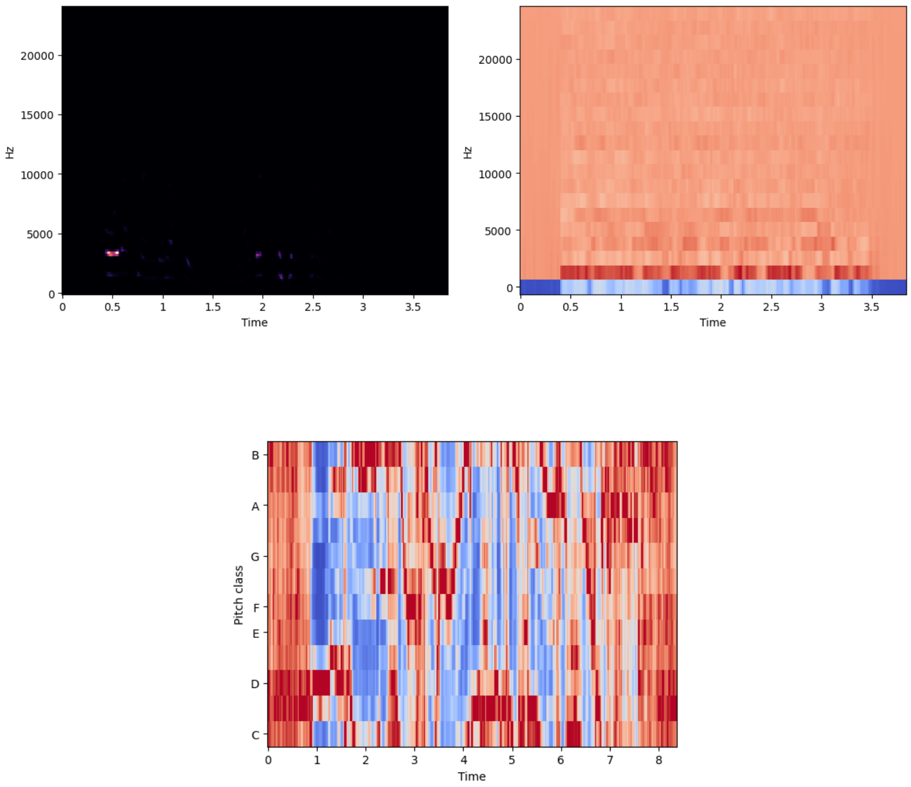

Figure 1: Mel-Spectrogram, MFCC and Chroma STFT

Each audio is divided into windows of two seconds without overlapping and for each window the three features are computed, then for each feature the mean value of each column is computed in order to obtain three vectors and lastly these three vectors are concatenated in a single vector with length of 160 elements.

Once that all the features are ready, the splitting in training, test and validation set is performed. The main approach used consists in a “naïve” splitting in which the features are divided: 70% for the training set, 20% for the test set and 10% for the validation set. Since there is no overlapping between windows the split can be performed randomly.

The training set is used to train the model, the test set is a completely independent subset of the dataset that the model has never seen during training or validation, and it is used to evaluate the final performance of the trained model. Lastly, the validation set that is a smaller subset of the dataset that is used to assess model performance during training and is exploited in the early stopping technique helpful to avoid overfitting.

2.2.2 Choice of the Model

During the project development different models are implemented and tested, including a model that analyze the mel-spectrogram using 2D convolutions. Subsequently only the results obtained with the top-performing model are discussed. The chosen model is composed by:

- Input Layer: It takes as input the features previously extracted.
- Convolutional Layers: They extract the relevant knowledge from the features provided to the model. There are five Convolutional 1D Layers, each of them alternated by a Max Pooling Layer. Note that the number of the filter increase while the kernel size decrease as going forward into the structure.
- Max Pooling Layers.
- Flatten Layer: It aggregates the features in a 1-dimensional vector.
- Fully Connected Layers: There are two dense layers for further processing of the extracted features, alternated by two dropout layers to prevent overfitting.
- Output Layer: it is a linear layer that consist in seven units and a Softmax activation function. The Softmax converts the output values into probabilities, representing the likelihood of the input belonging to each of the seven emotion classes.

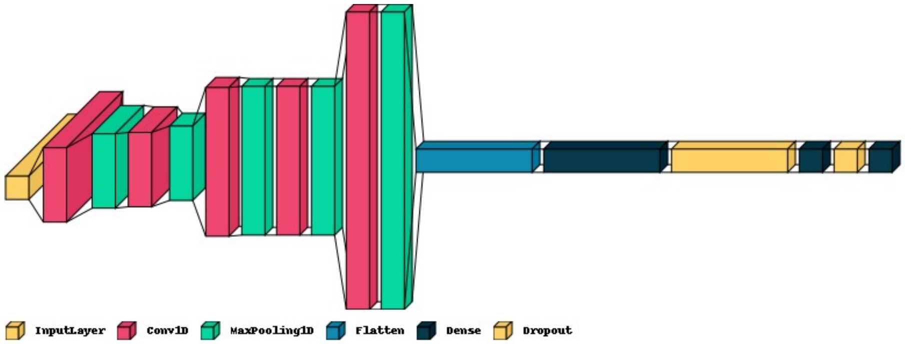

Figure 2: Structure of the implemented model

2.2.3 Training on EMOVO

In this section the model previously described is trained by using the extracted features from the EMOVO's signals. The model is trained with a total of 200 epochs, but an early stopping is added to prevent overfitting. When the validation loss stops improving or begins to worsen, the early stopping ensures that the training process is stopped before the model starts to overfit the training data.

Once the model is trained, it reaches an accuracy of 54% and by plotting the confusion matrix the results are the ones in (Fig. 3).

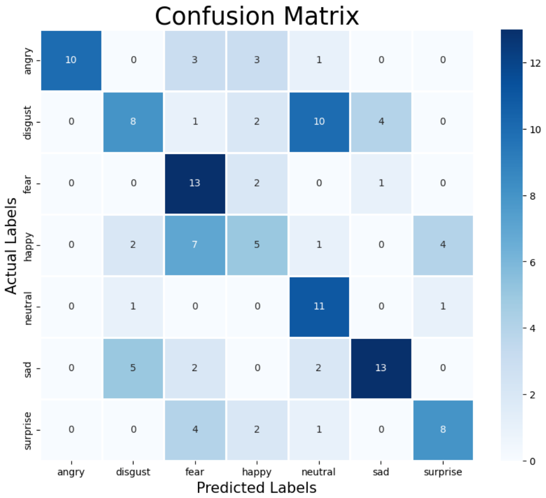

Figure 3: Confusion matrix of the model trained on EMOVO

2.2.4 Transfer Learning

At this point of the project the aim is to mitigate the data scarcity problem and improve the performance of the model trained on EMOVO. As mentioned in the introduction two approaches are explored and the first one is about transfer learning.

For this purpose two other datasets are used: RAVDESS and CREMA-D. Different from the EMOVO's case, firstly the splitting into training, test and validation is performed, using the "naive" approach, and secondly the different features are extracted. In this way there is no more the risk of sending two overlapping windows in two different sets, hence it is possible to extract more windows from the data by simply overlapping them.

Later the same model is trained using the features just extracted and the results obtained are the ones in (Fig. 4), reaching an accuracy of 39%.

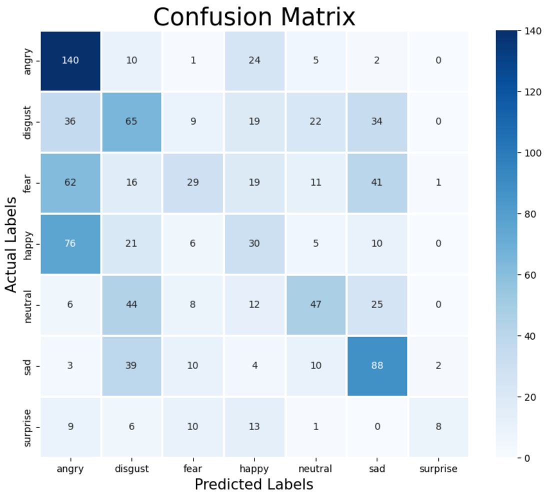

Figure 4: Confusion matrix of the model trained on RAVDESS and CREMA-D

This last model is also evaluated using the features extracted from EMOVO's signals, obtaining the relative confusion matrix (Fig. 5) with an accuracy of 30%. Clearly these results are due to the fact that the source and target domain have some huge differences.

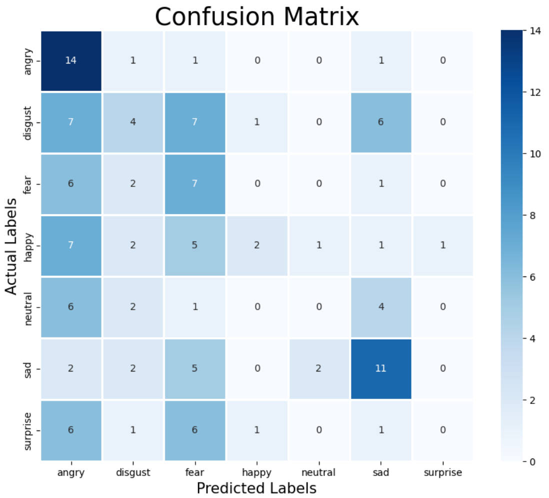

Figure 5: Confusion matrix of the model trained on RAVDESS and CREMA-D, evaluated on EMOVO

Now it is time to perform transfer learning through fine-tuning. In this process, the convolutional layers of the pre-trained model on RAVDESS and

CREMA-D are frozen, meaning their weights are not updated during training, while only the final linear layers are trainable.

After re-training the model with the EMOVO's features, the performance reaches an accuracy of 60% and the confusion matrix becomes:

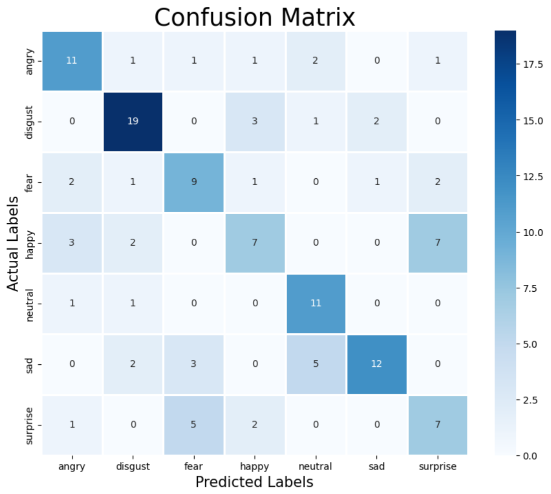

Figure 6: Confusion matrix of the model trained on EMOVO after applying transfer learning

2.2.5 Data Augmentation

The second approach used to deal with the data scarcity problem is data augmentation. The transformations applied to the EMOVO's audio signals are:

- Noise injection: Adding some random noise to the original data.
- Time stretch: Altering the temporal duration of the audio signal without changing its pitch (frequency).
- Pitch shift: Changing the pitch of the audio signal while maintaining its temporal duration.

By applying these transformations and a combination of them to the training data, the model becomes exposed to various data variation, making it more adaptable and accurate.

The parameters of these transformations are selected to generate augmented data that significantly differs from the original data, creating distinct training examples, while still retaining enough similarity to the originals for a human to recognize the original emotion when listening to them.

Now the splitting of training, test and validation data is performed by using three different methods:

- “Naïve” approach previously explained.
- Leave-One-Subject-Out (LOSO): One subject’s entire data is used as the test set, while the remaining subjects’ data are used for training.
- Leave-One-Phrase-Out (LOPO): One complete phrase is used as test set, while the rest of the data is used for training. In the case of this project the chosen phrase is one of the longest ones to still have enough samples inside the test set.

Finally, the model is trained for each of the three previously approaches, reaching an accuracy of 65%, 20% and 70% respectively.

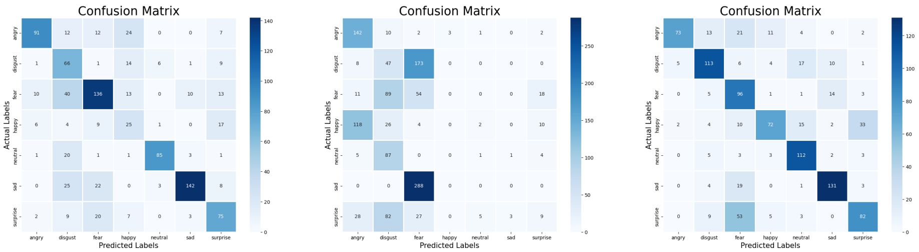

Figure 7: Confusion matrices for the ”naïve”, LOSO and LOPO approaches

3 Experiments

3.1 Training on EMOVO

The first result to comment is about the model trained with only the features extracted from the originals signals of EMOVO. From (Fig. 3) it is possible to see how the model performs better in predicting certain emotions (e.g., ”angry” and ”neutral”) while struggling with others (e.g., ”happy”, ”disgust”).

These results may be affected by class imbalance, where one or more emotions are not properly represented in the training set (this happen because the split is random and the dataset is small). The overall accuracy is therefore of 54% indicating that there is room for improvement in the model’s performance.

3.2 Transfer learning

The second step concerns applying transfer learning to overcome the data scarcity problem. At first the model is trained on Ravdess and CremaD datasets, but the results from (Fig. 3) are not so encouraging: the model performs reasonably well for the ”angry” and ”neutral” classes, but it is less satisfactory for the other classes. This is due to the fact that probably the model is too simple to perform a complex task on a dataset with an high quantity of speakers (It’s possible to get better result on Ravdess and CremaD by simply increasing the number of filter in the convolutional part of the model, but this will have a low effect on the evaluations made using Emovo). Anyway, this does not mean that the model doesn't learn some good features that can be usefull in a target domain.

In fact, by exploiting the knowledge previously learned and performing fine tuning on EMOVO the results are much better (Fig. 5), reaching an overall accuracy of 60%. The model performs relatively well for some classes like "disgust", "neutral" and "sad". However, the performance is less consistent for other classes, such as "happy" and "surprise".

3.3 Data Augmentation

Lastly the model is trained using data augmentation, the data are augmented as described before and the pre-processing part is rewritten in order to take into account the origin of data (file, actor and phrase) to allow us to perform different types of evaluations.

3.3.1 "Naïve" split

The first type consist in the "Naïve" split, in which the results are better than the ones obtained since now: the model performs well for "angry", "neutral", "sad", and "fear" emotions, but struggle with the other ones. The overall accuracy of the model is 0.65, which indicates that it correctly predicts emotions for 65% of the instances in the dataset.

3.3.2 Leave-One-Subject-Out

In the Leave-One-Out-Subject method, unfortunately, the accuracy drops to 20%: the model seems to have difficulties in predicting all emotions accurately. This is due to the fact that the model is learning features specifically from the actors that are in the dataset and so it means that if the model is used in a real life scenario, the data of the final user must be included into the dataset.

Another option can be training the model, or a more complex model, using datasets obtained by using multiple subjects. Since in this project a model is trained using more than 6 subjects (the model trained on RAVDESS and CREMA-D), it is possible to fine tune the model on the EMOVO dataset hoping that this last model has learned more subject invariant features. In addition, since this task is dealing with augmented data, it is decided to set one more layer in the trainable state.

The results are still trash and the model is able to classify correctly only 2 classes the accuracy has significantly increased from before reaching 32%.

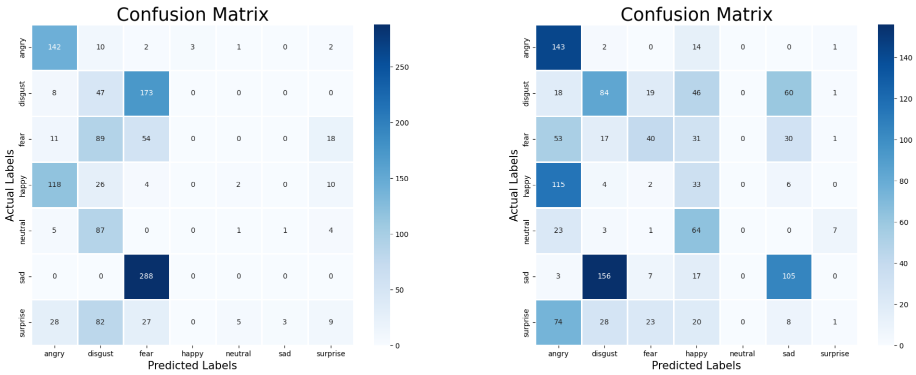

Figure 8: Confusion matrices for the LOSO approach without and with transfer learning

3.3.3 Leave-One-Phrase-Out

In this method the results are not worst, but are even better than the ones obtained with "naïve" split. There are two possible reasons why this happened:

- The model performances are good for some phrases and slightly worst for others.
- Splitting in this way will lead to an almost balanced train set (same number of samples for each class in the train set), thing that is not guaranteed with the "naïve" split.

3.4 Observation related to other models

During the development of the project different models were tested, one of them was really similar to the one presented except for two other features that were concatenated with the input vector: the mean of the *rms* and the mean of the *zcr*. This two features are uncorrelated with the labels and have no sense to be concatenated with time depend features inside the input of the CNN. In fact, with this two feature the result applied on only Emovo doesn't get accuracy values greater than 33%, anyway using transfer learning and data augmentation the Network is able to overcome this problem and results slightly below 60% are reached.
This mean that the results are worst than the ones of the presented model but using transfer learning and data augmentation leads to an huge improvement.

3.5 Overall observations

An important observation to add is that all the results from each task suffer from a high variance, as running the code multiple times it may lead to different results because not all random processes happening during training can be controlled using seeds.

4 Conclusions

Finally, let's recall the main results related to the EMOVO dataset. It is possible to see the gradual improvement, except for the LOSO approach, of the model's performance reaching a maximum of 70% with the LOPO approach.

		Data Augmentation		
Only EMOVO	Transfer Learning	"Naïve"	LOSO	LOPO
54%	60%	65%	20%	70%

Table 1: Summary of all the EMOVO's results

Furthermore, it is worth saying how all these approaches can be used, either in the same way or through different strategies, while also leveraging other datasets beyond just audio signals.

In conclusion, this project demonstrates that using CNNs, along with methods to deal with the data scarcity problem in small datasets, enables effective Speech Emotion Recognition.