Interaffection of Multiple Datasets with Neural Networks in Speech Emotion Recognition*

Ronnypetson S. da Silva[1] **, Valter Akira M. Filho**[1] **, Mario C. M. de F. Souza**[2]

[1]CPQD, Campinas, São Paulo, Brazil

{ronnypetson,valterf}@cpqd.com.br

[2]CPFL Energia, Campinas, São Paulo, Brazil

mariomotta@cpfl.com.br

Abstract. *Many works that apply Deep Neural Networks (DNNs) to Speech Emotion Recognition (SER) use single datasets or train and evaluate the models separately when using multiple datasets. Those datasets are constructed with specific guidelines and the subjective nature of the labels for SER makes it difficult to obtain robust and general models. We investigate how DNNs learn shared representations for different datasets in both multi-task and unified setups. We also analyse how each dataset benefits from others in different combinations of datasets and popular neural network architectures. We show that the long-standing belief of more data resulting in more general models doesn't always hold for SER, as different dataset and meta-parameter combinations hold the best result for each of the analysed datasets.*

1. Introduction

In the process of communication between people, speech not only carries language but also emotion, which is important in transmitting the state-of-mind of the speaker. Therefore, one way of enhancing speech processing systems and its various applications is by using the emotional information of the user. Speech Emotion Recognition (SER) is a field whose objective is to automatically infer emotion from speech signals. It has been an active field of research for at least over twenty years [Schuller 2018] and has many applications in the areas of affective computing, human-computer interaction and speech analytics [Ramakrishnan and El Emary 2013].

As with most problems that can be solved with data-driven techniques, Deep Learning has been recently applied to SER [Pandey et al. 2019], including a variety of architectures [Fayek et al. 2017, Lim et al. 2016, Trigeorgis et al. 2016], alternative optimization [Han et al. 2014], mixed supervision [Mao et al. 2014], attentive mechanisms [Mirsamadi et al. 2017, Neumann and Vu 2017, Sarma et al. 2018], and so on. Deep Learning holds the current state-of-the-art for SER and prior to its use the main approaches were based on traditional techniques in speech processing, such as hand-crafted features [El Ayadi et al. 2011] and Gaussian Mixture Models combined with classifiers like Support Vector Machines [Dileep and Sekhar 2013].

*Supported by ANEEL's R&D program (Project ID PD0063-3039/2018) in partnership with CPFL ENERGIA group companies.

Table 1. Summary of the emotional speech datasets.

	IEMOCAP	TESS	RAVDESS	CustomerPTBR
Speakers	10	2	24	> 30
Language	English	English	English	Brazilian Portuguese
Spontaneity	Scripted and acted	Scripted	Scripted	Natural
Utterances	10039	2800	7356	5364
Classes	9	8	8	5
Situation	dialogue	monologue	monologue	dialogue

Deep Learning models are expected to scale well with data availability. But even though many public datasets have been provisioned along the years of research in SER, it is difficult to use them in conjunction due to the each one applying its own annotation protocols. Many differences arise both in the nature of the annotation (dimensional or categorical), the subsequent choice of labels [Neumann and g. Thang Vu 2018], the number of annotators and level of agreement between them. For the most part, we believe this is because the nature of the problem is inherently subjective [Lee 2019], and so its formulation.

In the last decade some works on SER tried to address the poor generality of models to different domains [Lefter I. 2010, Schuller et al. 2010]. A special case is multilingual and cross-lingual SER [Lee 2019, Neumann and g. Thang Vu 2018]. The strategies include conditioning on language or domain [Sagha et al. 2016], removing cross-domain variations [Chiou and Chen 2014], and merging datasets [Neumann and g. Thang Vu 2018]. In this work we analyse three public datasets in English and a proprietary one in Brazilian Portuguese and their inter-affection with different modeling strategies.

2. Emotional datasets

We use the datasets IEMOCAP [Busso et al. 2008], TESS [Dupuis and Pichora-Fuller 2010], RAVDESS [Livingstone and Russo 2018], and a private dataset which we will call CUSTOMERPTBR. In Table 1 we compare the overall characteristics of the datasets. CustomerPTBR consists of recordings along with textual transcriptions and emotional labels from phone calls of customer services. The majority of the calls are between clients and attendants, although there are calls between attendants (e.g. transfer of attendance). The emotional labels present in this dataset are *happy*, *angry*, *neutral*, *sad*, and *fearful*.

Between those datasets, IEMOCAP and CustomerPTBR have more phonetic variation as they are richer in language than TESS and RAVDESS. Each of the 5 sections from IEMOCAP have distinctive dialogues and CustomerPTBR is naturally varied in this aspect. Meanwhile, utterances from TESS and RAVDESS consist only of a limited number of phrases that are repeated with different emotional modes and actors.

3. DNNs and Speech Emotion Recognition

Speech-related tasks in Machine Learning may have temporal dependencies in the order of hundreds or thousands of time steps due to the usually high sampling rate of audio. This

applies to SER (Speech Emotion Recognition), as the models should be able to aggregate information in a big temporal window in order to infer overall emotion. For such, it is common to use LSTM (Long Short-Term Memory) [Hochreiter and Schmidhuber 1997] and TDNN (Time-Delay Neural Networks) [Waibel et al. 1989] layers, although some recent work apply Self-Attention [Sarma et al. 2018, Vaswani et al. 2017] as temporal aggregation.

[Mirsamadi et al. 2017] evaluates the effect of weighted mean pooling where the weights come from what they call a *local attention* mechanism. Their network includes a LSTM and a linear transformation responsible for attributing each time step an attentional score (e.g. higher scores indicate higher importance of the time frame in the task). They evaluate the models only on the IEMOCAP dataset. They show that the attentional weighting with LSTM is better than regular mean without LSTM (just linear and activations).

[Sarma et al. 2018] investigate the learning of filters from the pure wave form in the temporal domain. Their network consists of interleaved TDNN and LSTM layers plus Network-in-Network (NiN) [Ghahremani et al. 2016] layers for raw signal processing and Self-attention. They compare the results with the frequency-domain representation and report that the filters learned give substantially superior results over the pre-defined filters that are used in the frequency-domain representation, like FFT (Fast Fourier Transform), and MFCC (Mel-frequency Cepstral Coefficients). Their best result surpasses [Mirsamadi et al. 2017] in both weighted accuracy (by 8.31%) and unweighted accuracy (by 4.37%). It supposedly defines the new state-of-the art on IEMOCAP.

3.1. DNNs with multiple datasets

[Neumann and g. Thang Vu 2018] show that it is possible to merge English (IEMOCAP) and French speech (Recola [Ringeval et al. 2013]) to obtain one model almost as good as one model for each dataset. They also show that models trained on IEMOCAP can be fine-tuned for Recola and vice-versa with results better than pure cross-language (trained in one and tested in the other language). [Lee 2019] applies dropout regularization on a multi-task setup with English (IEMOCAP) and Japanese (JTES [Takeishi et al. 2016]) speech. The results show moderate gains for both datasets. Differently from our work, it feeds the network with emotional descriptors and their tasks are not defined in terms of recognition in each dataset, but in classification of emotion, gender, and language. In the cases with only emotion recognition for all the data merged, they use just one regression layer, like in our "unified" setup.

[Zhang et al. 2017] defines a multi-task setup similar to ours, where each task is the emotion recognition in each dataset. Their setting includes nine smaller datasets and they do not variate the configuration of their DNN, which is composed solely of fully-connected layers. They also use the learned shared representations in order to train single-task models. In turn, those re-trained single-task models show similar performance to the multi-task model trained a priori. We leverage the four datasets despite their different formats by creating a shared representation scheme where a Deep Neural Network (DNN) learns a latent feature map that is used separately for the classification with respect to each dataset. If the output labels of each dataset share similarities (e.g. it is possible to map classes in one dataset to classes in other), we can even have a single classification layer.

4. Experimental setup

In this section we present all the experimental steps, including data preparation and augmentation, feature extraction and our choice of models and hyper-parameters for optimization.

4.1. Choice of emotional classes

Given the emotional classes from the datasets, we select a common subset of those classes that contain a minimum number of utterances. This is specially necessary for the unified case (equivalent classes from different datasets mapped into one class), and although it is not strictly necessary for the multi-task case (equivalent classes remain separated by dataset) we keep this selection in favor of comparability across datasets. Those classes are "happy", "angry", "neutral", and "sad" (the same as in [Sarma et al. 2018] and [Mirsamadi et al. 2017]).

4.2. Data split, processing and augmentation

We split each dataset in train, validation, and test sets without speaker overlap. IEMOCAP is already divided in 5 sections, each one with two different speakers - one male and one female. We then take 3 sections for training, 1 for validation, and 1 for test. RAVDESS is also separated by speakers and we divide it in the proportion of 70% utterances for training, 20% for validation, and 10% for test. The division of CustomerPTBR goes along the same line of RAVDESS. Because TESS only has two speakers, it is just used in training and validation.

The input features of all models are the log-spectra of short-time Fourier transforms (STFTs) from the input signals (Fig. 1). The size of the FFT window is 512 samples at 8000 samples per second, with a window hop of 128 samples. This results in a sequence of 257-dimensional vectors for each example. In order to account for the difference in the number of examples in each dataset, we perform data augmentation in proportions such that the augmented datasets end up with similar number of examples. The augmentations consist of additive background noise [Barker et al. 2015], VTLP (Vocal Tract Length Perturbation) [Ko et al. 2015], RIR (Room Impulse Response) [Jeub et al. 2009, Kinoshita et al. 2013, Ko et al. 2017, Nakamura et al. 2000, Scheibler et al. 2018], MP3 codec simulation (encoding-decoding), band-pass filtering, and volume perturbation.

4.3. DNN architectures and variations

Our neural networks are similar to the LSTM plus pooling setup in [Mirsamadi et al. 2017], although we vary the block of layers before Local Attention. We also vary the statistical aggregation of the pooling layer by testing aggregation by weighted mean and variance besides just weighted mean. The LSTM layers have hidden dimension of 128 and the dimension of the hidden linear layer is 512 (the same as in the TDNN setup). Dropout rate is 0.5 for all configurations and there is dropout after the recurrent layers. Without considering the difference in the last layer between the unified and multi-task setup (the multi-task setup has one classification "head" for each dataset, each one having the same input from the shared part of the network), we test the six combinations of temporal block and statistical pooling. We define all the neural network steps, from pre-processing to temporal blocks, aggregation, and classification in figures 1 to 5 in this order. Note that the temporal blocks are not used together, but each one in its configuration.

4.3.1. Pooling with Local Attention

A key part of the statistical pooling are the attentional scores given to each time-step by the Local Attention layer [Mirsamadi et al. 2017]:

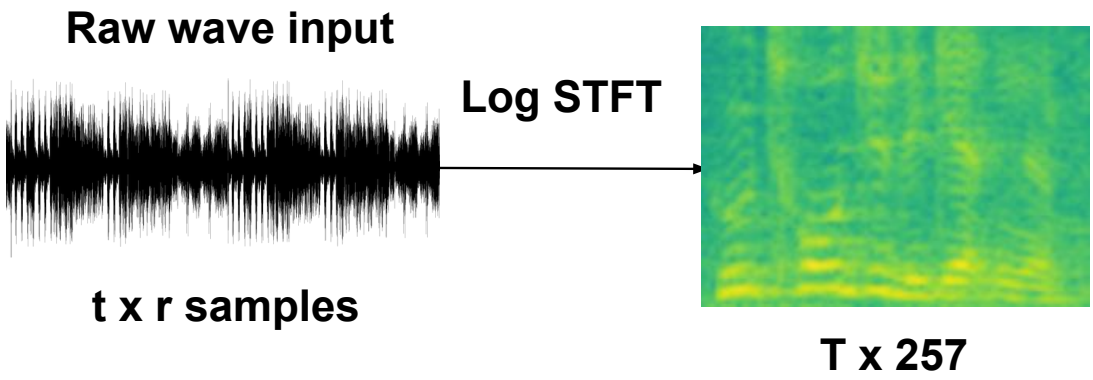

Figure 1. Pre-processing block that feeds all of the DNNs presented. t is the time duration in seconds and r is the sample rate. T is the resulting time-related dimension of the spectral representation.

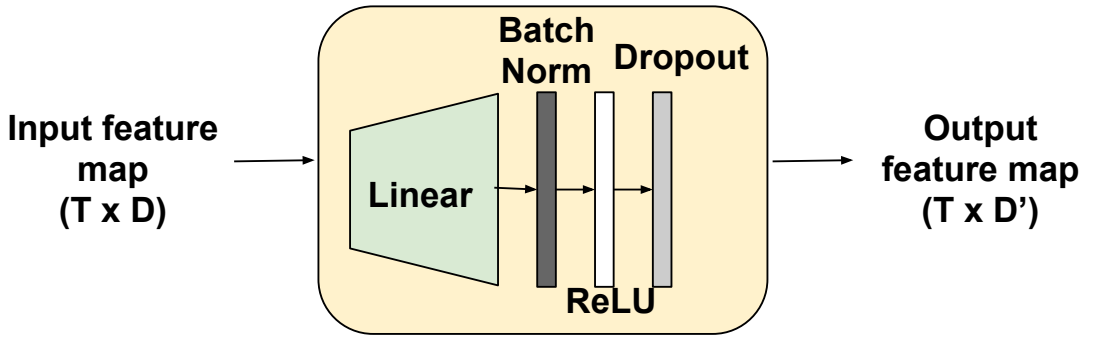

Figure 2. "Preparation" block inside the temporal blocks with LSTM and BLSTM. The temporal block with TDNN instead does not have this component, as its input comes directly from the pre-processing block. T denotes the time dimension. D and D' are the input and output dimensions of the time "slices".

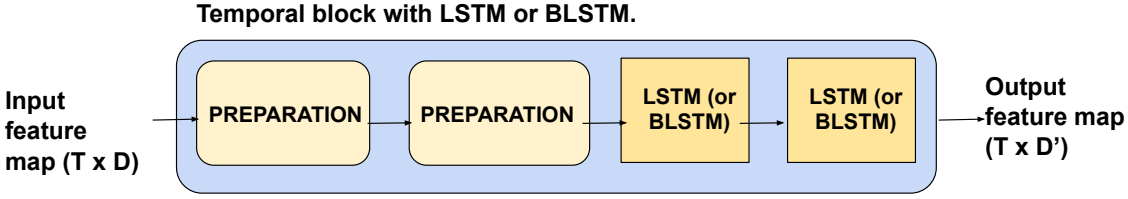

Figure 3. "Temporal" recurrent blocks that receive input from the pre-processing and whose output feed the Local Attention plus aggregation layer.

$$\alpha_t = \frac{exp(\mathbf{u}^T \mathbf{x}_t)}{\sum_{\tau=1}^{T} exp(\mathbf{u}^T \mathbf{x}_\tau)} \tag{1}$$

where α_t is the score for instant t, $\mathbf{u}$ is a vector of parameters, and $\mathbf{x}_t$ is the input feature map at instant t coming from the previous layer. We then interpret α as a probability distribution over the time-steps and we can reformulate Equation 2 from [Mirsamadi et al. 2017] as the expected value of $\mathbf{x}$ with respect to the attention weights:

$$\mathbb{E}[X] = \sum_{\tau=1}^{T} \alpha_\tau \mathbf{x}_\tau \tag{2}$$

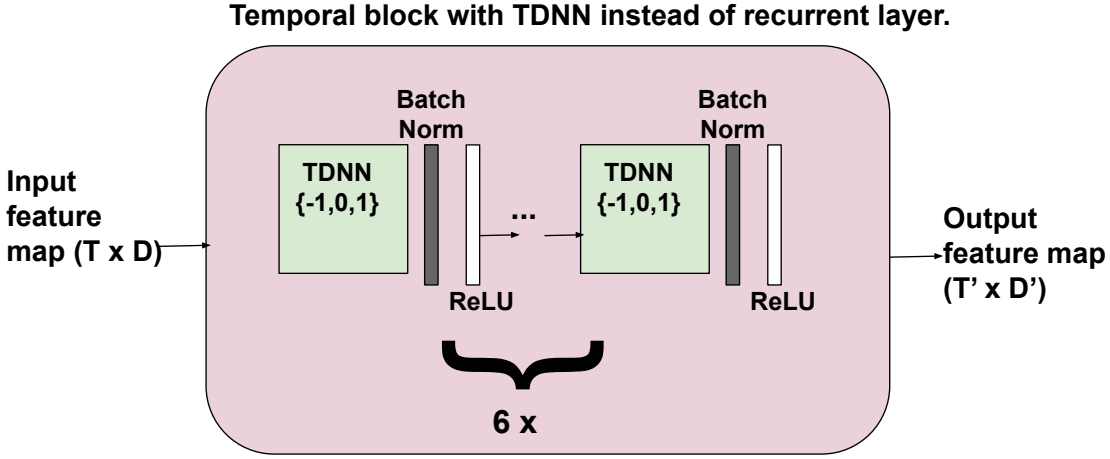

Figure 4. "Temporal" TDNN block with the same purpose of the recurrent ones. {-1, 0, 1} denotes the context window of the TDNN sliding filters, that is the same context window for the 6 TDNN layers in this block.

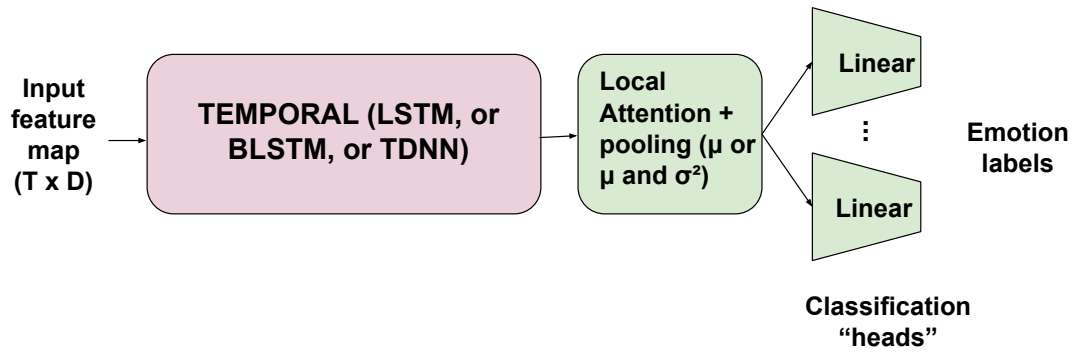

Figure 5. Temporal block followed by local attention aggregation and then by the classification "heads", that can be just one or as much as the number of datasets depending on the task type. The Local Attention with aggregation block is defined as in equations 1, 2, and 3.

and the variance is derived as follows:

$$\mathbb{V}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2 = \sum_{\tau=1}^{T} \alpha_\tau \mathbf{x}_\tau^2 - (\sum_{\tau=1}^{T} \alpha_\tau \mathbf{x}_\tau)^2 \tag{3}$$

In this formulation, the original Local Attention is composed only by the expectation of the previous layer. The variance is an added novelty to which we compare the original in the our experiments.

4.4. Training

The training is based on mini-batch gradient descent iterations, where the updates with the mini-batches of each dataset are alternated. We apply gradient clipping of 10 and weight decay [Krogh and Hertz 1992] as regularization. The optimizer Adam [Kingma and Ba 2014] with initial learning rate $3\mathrm{e}{-4}$. The criterion for stop training is patience of 20 epochs of 100 steps for improvement of at least 0.2% in the loss, that is the cross-entropy weighted by the number of examples in each class (one set of weights for each dataset).

We had 14 different dataset combinations which, when combined with the 12 different meta-parameter configurations (3 temporal block types, 2 aggregation layer types and 2 task setups) resulted in 168 experiments in total. Experiments that combined more than one dataset with a test set (that is, excluding the TESS dataset) generated results for all test sets, which means the 168 experiments resulted in actually 288 results, summarized in Section 5.

5. Results

In this section, we present the results for each dataset in terms of weighted accuracy (WA) and unweighted accuracy (UA). We analyse how they are affected by each set of meta-parameters that define the neural network, and by the combination of datasets. Due to the aforementioned large amount of experiments, we generate summaries with the proportion of cases in which each network configuration or dataset combination had the best result.

5.1. Best results by dataset

We show the best configuration given target dataset in Table 2 according to WA and UA separately. The gain for CustomerPTBR from its best configuration with respect to the best isolated training is notably substantial, being 8.3% in WA and 18.8% in UA. In the case of RAVDESS, this gain is more modest, but still exists for both WA and UA. IEMOCAP did not improve from its best single model.

Aside for the multi-task setup, which dominates the best scenarios over unified, those best results do not indicate a clear best set of meta-parameters.

5.2. By model meta-parameters

Due to the limitations in analysing the meta-parameters in Table 2, we present in Table 3 the proportion of best results for each meta-parameter configuration when applied over all the dataset combinations for each of the 3 target test sets. Both for WA and UA the TDNN

Table 2. Best configuration by target dataset according to WA and UA. When identifying the combinations, I stands for IEMOCAP, T for TESS, R for RAVDESS, and C for CustomerPTBR.

	Test Target	Comb.	Task	Temp.	Pooling	Best	Single
WA	**IEMOCAP**	I	-	BLSTM	$\mu\sigma^2$	58.8%	-
	RAVDESS	T,R	multi	BLSTM	$\mu\sigma^2$	81.3%	-
	CustomerPTBR	I,T,C	multi	TDNN	μ	60.4%	-
UA	**IEMOCAP**	I,T	multi	TDNN	$\mu\sigma^2$	59.4%	-
	RAVDESS	T,R	multi	BLSTM	$\mu\sigma^2$	81.9%	-
	CustomerPTBR	T,R,C	multi	TDNN	μ	83.5%	-
WA	**IEMOCAP**	-	-	BLSTM	$\mu\sigma^2$	-	58.8%
	RAVDESS	-	-	BLSTM	$\mu\sigma^2$	-	79.2%
	CustomerPTBR	-	-	BLSTM	$\mu\sigma^2$	-	52.1%
UA	**IEMOCAP**	-	-	BLSTM	$\mu\sigma^2$	-	57.7%
	RAVDESS	-	-	TDNN	$\mu; \mu\sigma^2$	-	79.2%
	CustomerPTBR	-	-	BLSTM	μ	-	64.7%

Table 3. Proportion of cases where each network was superior given the task type and metric (WA or UA).

	Task Setup	LSTM		BLSTM		TDNN	
		μ	$\mu\sigma^2$	μ	$\mu\sigma^2$	μ	$\mu\sigma^2$
WA	**Unified**	0%	12.50%	4.17%	4.17%	0%	8.33%
	Multi-task	0%	12.50%	16.67%	4.17%	12.50%	**25.00%**
UA	**Unified**	4.17%	4.17%	4.17%	0%	0%	16.67%
	Multi-task	4.17%	8.33%	12.50%	0%	8.33%	**37.50%**

temporal block with $\mu\sigma^2$ pooling and multi-task setup had the biggest share of cases with best results,

In Table 4, instead of counting for all 12 meta-parameter permutations, we compare each one individually. Initially, it corroborates the previous analysis, though it seems to indicate that the pooling type and the task setup are more consistent in terms of gains. This helps explaining why in Table 2 there are more BLSTM best results than TDNN. Both the $\mu\sigma^2$ pooling and the multi-task setup incur in more parameters to the trained model, though training and inference times had no significant difference. A final interesting thing to notice is that BLSTMs did not show consistent superior performance to their unidirectional counterpart, even though there are no best LSTM results in Table 2.

5.3. By dataset combination

In Table 5, we count how many times a dataset was used (column) in the best result of a test set (row). Each of the 12 possible meta-parameter permutations had a best result with a certain combination of datasets, from which we derive the percentage of each cell.

Table 4. Proportion of cases where each meta-parameter was superior for each metric.

	By temporal block			By pooling type		By task setup	
	LSTM	**BLSTM**	**TDNN**	μ	$\mu\sigma^2$	**uni**	**multi**
WA	29.17%	25.00%	**45.83%**	20.83%	**79.17%**	29.17%	**70.83%**
UA	16.67%	20.83%	**62.50%**	16.67%	**83.33%**	29.17%	**70.83%**

Table 5. Proportion of cases where each dataset was present in the best mix for each metric.

	Test Target	CustomerPTBR	IEMOCAP	RAVDESS	TESS
WA	**CustomerPTBR**	-	58.3%	41.7%	**66.7%**
	IEMOCAP	0%	-	**41.7%**	16.7%
	RAVDESS	0%	0%	-	**50,0%**
	All	0%	29.2%	41.7%	**44.4%**
UA	**CustomerPTBR**	-	25.0%	50.0%	**58.3%**
	IEMOCAP	16.7%	-	**58.3%**	41.7%
	RAVDESS	0%	0%	-	**50.0%**
	All	8.3%	12.5%	**54.2%**	50.0%

Since the usage of each dataset in the columns is non-exclusive, values in each cell may range from 0% to 100%. Roughly, we may interpret that, if the value from a certain cell is above 50%, the column dataset helps the row dataset in its classification task.

Against our initial intuition, the dataset which appears more frequently in the best results for other datasets is the TESS dataset, which is the smallest and comprised of very restricted examples. We believed that, due to its size and higher variability, the IEMOCAP dataset would help other datasets the most, but it is still surpassed by the RAVDESS dataset in this regard.

Our custom dataset helped others the least, possibly due to its language mismatch, but since the IEMOCAP dataset also struggles to ameliorate the accuracy of other datasets, we also hypothesize that the supposed domain variability from both datasets actually hinders the ability to combine them. This may again be due to the inherent subjective nature of the SER task.

6. Conclusion

We experimented with multiple emotional speech datasets applying different neural network topologies, generating results with multiple recombination of the former and meta-parameters of the latter. We showed that not always more data results in better models in SER, probably due to its inherent subjective nature. An extension of this study would include more corpora. It could narrow the search of meta-parameters to the best ones highlighted in this work. This would help us focus on the inter-affection between datasets. We could also explore more elements of Deep Neural Networks like Self-attention and NiN

layers, or other feature extractors. Also, it would be interesting to explore more datasets that are biased, small, and from different domains in order to check the robustness of shared representations.

Acknowledgments

We thank *CPQD* and the *CPFL* group for the funding and support of this work, which is part of project P&D ANEEL (PD-00063-3039/2018).

References

Barker, J., Marxer, R., Vincent, E., and Watanabe, S. (2015). The third 'chime'speech separation and recognition challenge: Dataset, task and baselines. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 504–511. IEEE.

Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S., and Narayanan, S. S. (2008). Iemocap: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42(4):335.

Chiou, B.-C. and Chen, C.-P. (2014). Speech emotion recognition with cross-lingual databases. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 558–561.

Dileep, A. D. and Sekhar, C. C. (2013). Gmm-based intermediate matching kernel for classification of varying length patterns of long duration speech using support vector machines. *IEEE Transactions on Neural Networks and Learning Systems*, 25(8):1421–1432.

Dupuis, K. and Pichora-Fuller, M. K. (2010). *Toronto emotional speech set (TESS)*. University of Toronto, Psychology Department. Available in `https://tspace.library.utoronto.ca/handle/1807/24487`.

El Ayadi, M., Kamel, M. S., and Karray, F. (2011). Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognition*, 44(3):572–587.

Fayek, H., Lech, M., and Cavedon, L. (2017). Evaluating deep learning architectures for speech emotion recognition. *Neural Networks*, 92.

Ghahremani, P., Manohar, V., Povey, D., and Khudanpur, S. (2016). Acoustic modelling from the signal domain using cnns. In *Interspeech*, pages 3434–3438.

Han, K., Yu, D., and Tashev, I. (2014). Speech emotion recognition using deep neural network and extreme learning machine. In *Fifteenth annual conference of the international speech communication association*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Jeub, M., Schafer, M., and Vary, P. (2009). A binaural room impulse response database for the evaluation of dereverberation algorithms. In *2009 16th International Conference on Digital Signal Processing*, pages 1–5. IEEE.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kinoshita, K., Delcroix, M., Yoshioka, T., Nakatani, T., Habets, E., Haeb-Umbach, R., Leutnant, V., Sehr, A., Kellermann, W., Maas, R., et al. (2013). The reverb challenge: A common evaluation framework for dereverberation and recognition of reverberant speech. In *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 1–4. IEEE.

Ko, T., Peddinti, V., Povey, D., and Khudanpur, S. (2015). Audio augmentation for speech recognition. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Ko, T., Peddinti, V., Povey, D., Seltzer, M. L., and Khudanpur, S. (2017). A study on data augmentation of reverberant speech for robust speech recognition. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5220–5224. IEEE.

Krogh, A. and Hertz, J. A. (1992). A simple weight decay can improve generalization. In *Advances in neural information processing systems*, pages 950–957.

Lee, S. (2019). The generalization effect for multilingual speech emotion recognition across heterogeneous languages. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5881–5885.

Lefter I., Rothkrantz L.J.M., W. P. v. L. D. (2010). Emotion recognition from speech by combining databases and fusion of classifiers — springerlink.

Lim, W., Jang, D., and Lee, T. (2016). Speech emotion recognition using convolutional and recurrent neural networks. In *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 1–4.

Livingstone, S. R. and Russo, F. A. (2018). The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english. *PloS one*, 13(5).

Mao, Q., Dong, M., Huang, Z., and Zhan, Y. (2014). Learning salient features for speech emotion recognition using convolutional neural networks. *IEEE Transactions on Multimedia*, 16(8):2203–2213.

Mirsamadi, S., Barsoum, E., and Zhang, C. (2017). Automatic speech emotion recognition using recurrent neural networks with local attention. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2227–2231. IEEE.

Nakamura, S., Hiyane, K., Asano, F., Nishiura, T., and Yamada, T. (2000). Acoustical sound database in real environments for sound scene understanding and hands-free speech recognition. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*.

Neumann, M. and g. Thang Vu, N. (2018). Cross-lingual and multilingual speech emotion recognition on english and french. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5769–5773.

Neumann, M. and Vu, N. T. (2017). Attentive convolutional neural network based speech emotion recognition: A study on the impact of input features, signal length, and acted speech. *CoRR*, abs/1706.00612.

Pandey, S. K., Shekhawat, H. S., and Prasanna, S. R. M. (2019). Deep learning techniques for speech emotion recognition: A review. In *2019 29th International Conference Radioelektronika (RADIOELEKTRONIKA)*, pages 1–6.

Ramakrishnan, S. and El Emary, I. M. (2013). Speech emotion recognition approaches in human computer interaction. *Telecommunication Systems*, 52(3):1467–1478.

Ringeval, F., Sonderegger, A., Sauer, J., and Lalanne, D. (2013). Introducing the recola multimodal corpus of remote collaborative and affective interactions. In *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pages 1–8.

Sagha, H., Matějka, P., Gavryukova, M., Povolny, F., Marchi, E., and Schuller, B. (2016). Enhancing multilingual recognition of emotion in speech by language identification. *Interspeech 2016*, pages 2949–2953.

Sarma, M., Ghahremani, P., Povey, D., Goel, N. K., Sarma, K. K., and Dehak, N. (2018). Emotion identification from raw speech signals using dnns. In *Interspeech*, pages 3097–3101.

Scheibler, R., Bezzam, E., and Dokmanić, I. (2018). Pyroomacoustics: A python package for audio room simulation and array processing algorithms. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 351–355. IEEE.

Schuller, B., Vlasenko, B., Eyben, F., Wöllmer, M., Stuhlsatz, A., Wendemuth, A., and Rigoll, G. (2010). Cross-corpus acoustic emotion recognition: Variances and strategies. *IEEE Transactions on Affective Computing*, 1:119–131.

Schuller, B. W. (2018). Speech emotion recognition: two decades in a nutshell, benchmarks, and ongoing trends. *Communications of the ACM*, 61(5):90–99.

Takeishi, E., Nose, T., Chiba, Y., and Ito, A. (2016). Construction and analysis of phonetically and prosodically balanced emotional speech database. In *2016 Conference of The Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Techniques (O-COCOSDA)*, pages 16–21.

Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., and Zafeiriou, S. (2016). Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5200–5204. IEEE.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., and Lang, K. J. (1989). Phoneme recognition using time-delay neural networks. *IEEE transactions on acoustics, speech, and signal processing*, 37(3):328–339.

Zhang, Y., Liu, Y., Weninger, F., and Schuller, B. (2017). Multi-task deep neural network with shared hidden layers: Breaking down the wall between emotion representations. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4990–4994.