Using Mozilla's DeepSpeech to Improve Speech Emotion Recognition

Andrejaana Andova
Jožef Stefan International Postgraduate School
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
andrejaana.andova@ijs.si

Stefano Bromuri
Open University of the Netherlands
Heerlen, Netherlands
Stefano.Bromuri@ou.nl

Mitja Luštrek
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
mitja.lustrek@ijs.si

ABSTRACT

A lot of effort in detecting emotions in speech has already been made. However, most of the related work was focused on training a model on an emotional speech dataset, and testing the model on the same dataset. A model trained on one dataset seems to provide poor results when tested on another dataset. This means that the models trained on publicly available datasets cannot be used in real-life applications where the speech context is different. Furthermore, collecting large amounts of data to build an efficient speech emotion classifier is not possible in most cases.

Because of this, some researchers tried using transfer learning to improve the performance of a baseline model trained on only one dataset. However, most of the works so far developed methods that transfer information from one emotional speech dataset into another emotional speech dataset.

In this work, we try to transfer parameters from a pre-trained speech-to-text model that is already widely used. Unlike other related work, which uses emotional speech datasets that are usually small, in this method we will try to transfer information from a larger speech dataset which was collected by Mozilla and whose main purpose was to transcribe speech.

We used the first layer from the DeepSpeech model as the basis for building another deep neural network, which we trained on the improvisation utterances from the IEMOCAP dataset.

KEYWORDS

speech emotion recognition, feature transfer, DeepSpeech

1 INTRODUCTION

There are many issues when trying to build a model for speech emotion recognition, but the main problem is the lack of emotional speech data. Collecting a dataset is often a challenging and effortful task, but in speech emotion recognition a few additional problems arise when creating a dataset. One of the main problems is that speech is a context-dependent problem. One could gather a dataset from job interviews and build a precise model that detects emotions in job applicants' speech. however, the same model would probably not work for a phone application that tries to analyze the emotions of its users. Thus, to build a general model for speech emotion recognition, one would need to gather a dataset composed of speeches used in different contexts, which is a hard task.

Most of the currently available emotional speech datasets are composed of actors performing scenes with different emotions. Finding actors and writing the scenes could be a costly and effortful task and, thus, it is hard to collect large amounts of data in this way. However, the major problem of this type of data is that all of the emotions are acted and may be more exaggerated when compared to real-life emotions [8]. This type of data is probably pretty different when compared to data from real-life applications where emotions are expressed with less intensity. To solve this problem, some researchers tried using transfer learning methods to build a model that is more robust to changes in the data.

Some researchers tried using speeches recorded in real-life scenarios and asked people to listen to these speeches and annotate the emotions they recognize in the speakers' voices. When collecting a dataset in this way one needs to find people that would listen to the whole dataset and annotate the data. The annotators would probably have different abilities to detect the emotions and different perceptions of what each emotion should be like. Because of this, in many cases not all of them will agree on which emotion is present in a sample. Another drawback of this type of data collection is that most of the time people do not experience extreme emotions. Because of this, such datasets will result in almost no emotions – the speech would be mostly neutral.

The main idea behind transfer learning is to use information from a dataset called source dataset to improve the performance of a target dataset. The source and the target datasets may have labeled or unlabeled data, may have the same data distribution or different data distribution, and they can be constructed to solve the same task or they may try to solve different tasks. Depending on this, there are different approaches to transfer learning. They are more thoroughly explained by S. J. Pan et al. [5].

In this work, we decided to follow the usual transfer learning approach, and use a pre-trained speech-to-text model trained on a large nonemotional English dataset collected by Mozilla. This model may not contain any emotional information that would be useful for our task, but we believe it contains information about the speech of the subjects that could be used in speech emotion recognition.

2 RELATED WORK

While research in speech emotion recognition where training and testing are done on one dataset has already been well-studied, using other datasets to make the model more generalized has been in focus only in recent years.

Information Society 2020, 5–9 October, 2020, Ljubljana, Slovenia

Table 1: Emotion distribution in IEMOCAP.

Anger	Happiness	Sadness	Neutral
500	94	467	392

Some researchers tried using unlabeled target data to improve speech emotion recognition models. Thus, Parthasarathy and Busso [6] connected supervised and unsupervised learning to improve the performance of speech emotion recognition on a target dataset. They used a network architecture similar to autoencoders to encode large amounts of unlabeled target data in an unsupervised way by putting the same speech in the input and the output of the network. To force the network to encode the emotional information from the speech, they connected the last encoding layer to another layer that was trying to learn the arousal, valence, and the dominance annotations on the speech in a supervised way. When they compared their method to other state-of-the-art models, it showed improvement in the arousal and the dominance space while in the valence space they got results slightly worse than the state-of-the-art.

Some authors thought about bringing the feature space from the source and the target data closer together. Thus, Song et al., [7] used MMDE optimization and dimension reduction algorithms to bring the feature spaces from the source and the target datasets closer together. After that, they used the shifted feature space from the source dataset to train an SVM model. They used the EmoDB dataset as a source dataset, and a Chinese emotional dataset collected by them as a target dataset. After they trained the SVM model on the source dataset only, they applied the model on the target dataset and showed that the model performed with 59.8% accuracy. These results show improvement when compared to an SVM model trained on the source dataset and tested on the target dataset without any dimension reduction applied, which performs with 29.8% accuracy. However, the best performance was achieved with a model trained and tested on the target dataset, which achieved 85.5% accuracy.

3 DATASET

In this research we used the Interactive emotional dyadic motion capture database (IEMOCAP) [1]. IEMOCAP consists of speech from ten different English-speaking actors (five male and five female), and it is the largest dataset for speech emotion recognition that we found publicly available. It consists of approximately twelve hours of data where actors perform improvisations or scripted scenarios, specifically selected to elicit emotional expressions. Since the actors were not given any specific emotions that they had to act, the database was annotated by multiple annotators into categorical labels, as well as dimensional labels, such as valence, activation, and dominance. The set of emotions the annotators could choose from was anger, happiness, excitement, sadness, frustration, fear, surprise, other, and neutral, but because most of the related work on transfer learning in speech emotion recognition only used anger, happiness, sadness and neutral utterances in their methods, we decided to also just use these emotions in our method.

We noticed that most of the time, the three annotators did not perceive the same emotion and, thus, we decided to eliminate all data where all three annotators did not agree on the detected emotion. This reduced the amount of data significantly. The distribution of the emotions after the data reduction is given in Table 1.

4 METHODOLOGY

We developed methods that transfer information from a large nonemotional speech dataset into a target emotional speech dataset. Since in most of the related work researchers were extracting information from smaller emotional speech datasets and transferring this information to other emotional speech datasets, this is the first attempt that we know of in which a transfer of information is tried from already well-defined pre-trained speech dataset into a smaller emotional speech dataset, which is the standard approach in most transfer learning applications.

However, to compare if the methods provide any useful improvement, we compare them to a baseline model that was trained and tested on IEMOCAP, and which does not use any kind of information transfer.

4.1 Baseline Model

To build a baseline classifier, we decided to use standard machine learning approaches trained on features extracted using OpenSMILE [2] as a baseline method. After testing several different machine learning approaches, we saw that Random Forest obtained the best results for most of the target datasets. Because of this, we decided to use a Random Forest classifier with 1000 trees and a maximal depth of 10 as a baseline model.

4.2 DeepSpeech Model

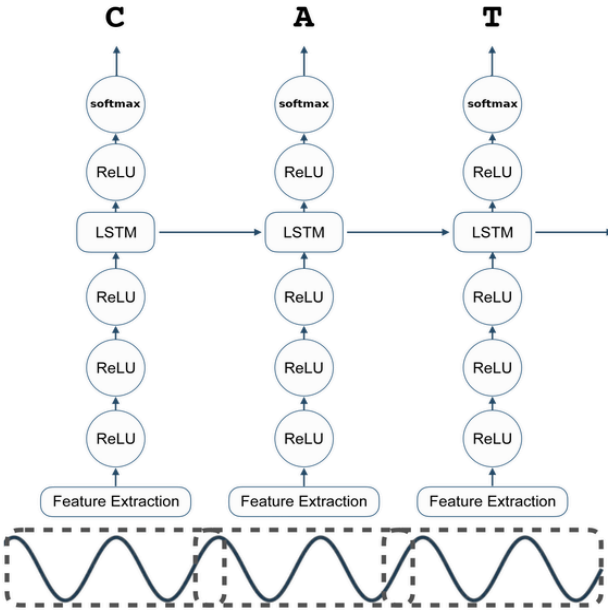

Figure 1: Architecture of the original DeepSpeech model.

DeepSpeech is a model that tries to provide transcriptions of a given speech. The model has been trained on the English

Table 2: Classification accuracy obtained from the majority classifier and baseline Random Forest Classifier compared to the DeepSpeech features method.

Model	Majority	Baseline	DeepSpeech features
Dense	34%	67%	58%
LSTM	34%	67%	7%
Dense1+Dense2	34%	67%	26%
Dense1+LSTM2	34%	67%	66%

data from the Mozilla Common Voice dataset [3]. This dataset consists of 1469 hours of speech data that has been recorded by 61521 different voices. The people whose voices were collected belonged to different nationalities (and thus different English accents), and different ages. All of this data is publicly available and can be easily accessed.

The architecture of the DeepSpeech model is presented in Figure 1. Each utterance is a time-series data, where every time-slice is a vector of MFCC audio features [4]. The goal of the network is to convert an input sequence x into a sequence of character probabilities for the transcription y.

The network is composed of five hidden layers. The first three layers are dense layers with 'ReLU' as an activation function. The fourth layer is an LSTM layer, the fifth layer is once again a dense layer with 'ReLU' activation function. The output layer has a softmax function which outputs character probabilities. In the example in Figure 1 the output of the first frame is the character 'C', the second frame outputs the character 'A', and the third frame outputs the character 'T', resulting with the word 'CAT'.

4.3 Transfer Learning Using DeepSpeech

We decided to experiment if we could transfer information from the DeepSpeech model that would be useful for the speech emotion recognition task. We used the representation learned by the DeepSpeech network to extract features for the IEMOCAP dataset. We used the output from the first layer in the DeepSpeech model as features for a given frame. We ended up with 2048 features for every 10-millisecond frame. So, if the whole utterance was 3 seconds long, we would receive a matrix with dimensions 1800x2048 after the deep speech feature extraction.

After the features from all the samples in IEMOCAP have been extracted, we trained a deep neural network using them. We simply added the layers from the new deep neural network on top of the first layer from the DeepSpeech model, and trained the new deep neural network from scratch by just using the samples from the IEMOCAP dataset. This way we repurpose the feature representations from the first layer of the DeepSpeech model.

We experimented with several different deep neural network architectures to see which one works best for this problem. In the first architecture, we used a feed-forward network on the extracted features per each frame. We used one hidden dense layer with 'relu' activation function and 204 neurons. We connected this layer to a dense layer with softmax activation function which predicted the emotion probabilities for each frame separately. Although in the IEMOCAP dataset there are no labels for each of the frames separately, we use the target label for the whole utterance as target label for each of the frames.

The second model architecture we tried was to use the features from the whole frame as input, and use a LSTM layer to learn the representations from the features. The LSTM layer is activated by a 'relu' function and has 20 hidden states. It is then connected to a dense layer activated by a 'softmax' activation function which predicts the label of the whole utterance.

The third network architecture is composed of two parts. In the first part we predict the emotion probabilities for each frame separately and in the second part we use the emotion probabilities predictions from the first layer to predict the emotion probabilities for the whole utterance. The first part of the architecture is the same as in the first network architecture and is trained one one half of the training data. In the second part of this network, we use the predictions from the first part as input to a dense layer with a softmax activation function. The second part of the network is trained on the other half of the training data. In this network architecture, for each sequence of 20 frames we predict one vector of emotions.

The fourth network consists of two separate parts and is presented in Figure 2. The first part takes the output of the DeepSpeech model, and tries to predict the probability for each of the target emotions separately. The first dense layer has a 'relu' activation function and outputs 204 features. It is then connected to another dense layer with a softmax activation function that predicts the emotions present in each frame separately. The second part of the network uses the output emotion probabilities from the first part of the layer as an input. The second part of the network consists of one LSTM layer which is trained on the second half of the training data. The LSTM layer is activated by a 'relu' function and has 20 hidden states. It is then connected to a dense layer activated by a 'softmax' activation function which predicts the label of the whole utterance. This network architecture in a way is a combination from the first and the second network architecture.

5 RESULTS

Since the DeepSpeech model is capable of learning language phases in the speech, we decided to remove all scripted utterances from the IEMOCAP dataset and use just the utterances in which the actors were asked to improvise. To evaluate the neural network architectures we used the leave-one-subject-out cross validation.

In Table 2 we present the results obtained from each of the deep neural network architectures that we tried as well as the accuracy of the baseline model and the majority classifier. In the results we can see that the LSTM network architecture that we tried performs quite poor, with classification accuracy of only 7%. The most probable explanation for this is that this architecture is quite complex since it has 2048 features for each frame, and it tries to train an LSTM model on all of these features. To train a model with this amount of parameters, we would need much more samples than the IEMOCAP improvisations.

The architecture that provides the best results is the one that uses a FFN to predict the features in each frame, and then uses a

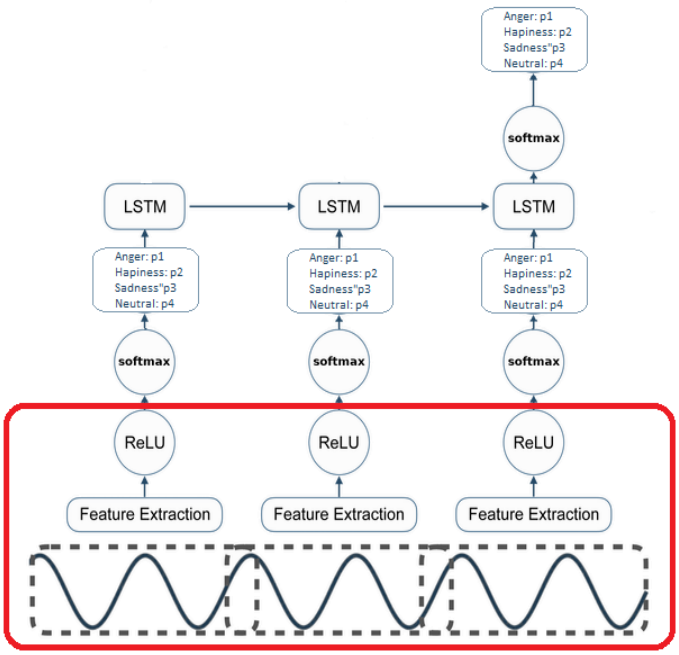

Figure 2: Architecture of the original DeepSpeech model.

LSTM network to predict the final emotion predictions for the whole utterance. We further experimented with this network architecture to see how much the length of the frames changes the performance of the model. The results are presented in Figure 3. In this figure, we can notice that the performance of the model can be improved by using bigger frames when training the LSTM part of the DeepSpeech model. However, the performance of the model does not differ a lot – only a few percentage points.

The results show that some of the DeepSpeech architectures can perform better than the majority classifier but none of the architectures outperforms the baseline model. A possible explanation for this could be that these two tasks are simply not related enough and we cannot use information from the DeepSpeech model to improve the performance of a model for speech emotion recognition.

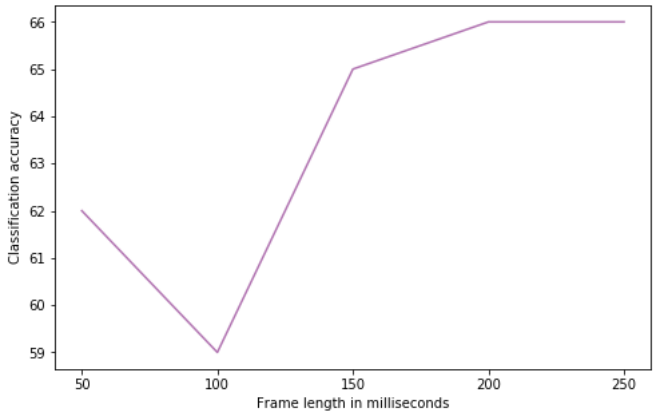

Figure 3: Performance of DeepSpeech model by using different frame lengths.

6 CONCLUSION

In this work we tried to improve a baseline speech emotion recognition classifier by transferring information from a pre-trained model. Although this transfer learning method has been most widely used in other computer science fields, most of the related work in speech emotion recognition developed transfer learning methods that transfer information from other emotional speech datasets into a target emotional speech dataset.

The pre-trained model we used was Mozilla's DeepSpeech that was developed as a speech-to-text model. To recognize emotions in speech, we used the first layer from the DeepSpeech model, on top of which we added a new classifier that was trained from scratch on an emotional speech dataset. This way we repurposed the feature maps learned previously for the dataset.

The results from this approach did not seem to improve the classification accuracy of the improvisations part in the IEMOCAP dataset. A possible explanation for this could be that the speech-to-text and speech emotion recognition tasks are simply not sufficiently related, and because of this the model could not extract any useful information from the DeepSpeech model. However, since this was the first attempt to transfer information from a well-defined pre-trained model to a speech emotion recognition task, we believe it is still a valuable attempt.

7 ACKNOWLEDGMENTS

This research has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No 769765

REFERENCES

[1] Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. 2008. Iemocap: interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42, 4, 335.

[2] Florian Eyben, Martin Wöllmer, and Björn Schuller. 2010. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, 1459–1462.

[3] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. 2014. Deep speech: scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*.

[4] Beth Logan et al. 2000. Mel frequency cepstral coefficients for music modeling. In *Ismir*. Volume 270, 1–11.

[5] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22, 10, 1345–1359.

[6] Srinivas Parthasarathy and Carlos Busso. 2019. Semi-supervised speech emotion recognition with ladder networks. *arXiv preprint arXiv:1905.02921*.

[7] Peng Song, Yun Jin, Li Zhao, and Minghai Xin. 2014. Speech emotion recognition using transfer learning. *IEICE TRANSACTIONS on Information and Systems*, 97, 9, 2530–2532.

[8] Carl E Williams and Kenneth N Stevens. 1972. Emotions and speech: some acoustical correlates. *The Journal of the Acoustical Society of America*, 52, 4B, 1238–1250.