In: *International Conference on Artificial Neural Networks (ICANN)*, Volume LNCS 12396, pages 529-540, doi:10.1007/978-3-030-61609-0_42

Variational Autoencoder with Global- and Medium Timescale Auxiliaries for Emotion Recognition from Speech*

Hussam Almotlak, Cornelius Weber, Leyuan Qu, and Stefan Wermter

University of Hamburg - Dept. of Informatics, Knowledge Technology
https://www.inf.uni-hamburg.de/en/inst/ab/wtm/
{8almotla,weber,qu,wermter}@informatik.uni-hamburg.de

Abstract. Unsupervised learning is based on the idea of self-organization to find hidden patterns and features in the data without the need for labels. Variational autoencoders (VAEs) are generative unsupervised learning models that create low-dimensional representations of the input data and learn by regenerating the same input from that representation. Recently, VAEs were used to extract representations from audio data, which possess not only content-dependent information but also speaker-dependent information such as gender, health status, and speaker ID. VAEs with two timescale variables were then introduced to disentangle these two kinds of information from each other. Our approach introduces a third, i.e. medium timescale into a VAE. So instead of having only a global and a local timescale variable, this model holds a global, a medium, and a local variable. We tested the model on three downstream applications: speaker identification, gender classification, and emotion recognition, where each hidden representation performed better on some specific tasks than the other hidden representations. Speaker ID and gender were best reported by the global variable, while emotion was best extracted when using the medium. Our model achieves excellent results exceeding state-of-the-art models on speaker identification and emotion regression from audio.

Keywords: Unsupervised learning · Feature extraction · Variational Autoencoders · VAE with auxiliary variables · Multi-timescale neural network · Speaker identification · Emotion recognition.

1 Introduction

Autoencoders have shown an obvious ability to capture the essential features in the data and reduce its dimensionality [4]. They are also, with their non-linear behavior, preferred for dimensionality reduction over other linear methods such as principal component analysis (PCA) when the data is very high dimensional.

* Supported by Novatec Consulting GmbH, Dieselstrasse 18/1, D-70771 Leinfelden Echterdingen and by the German Research Foundation (DFG) under the transregio Crossmodal Learning TRR-169.

Lately, much effort has been made on the processing of static data like images, and less on the processing of sequential data like audio and videos [12]. Sequential data typically holds instantaneous as well as long-term information. The longterm information in the speech describes specific characteristics from the voice. Some of these characteristics do not change with time, such as health status, age, gender, and speaker ID. Others could change with time like the emotion. This work pays attention to both kinds and aims to preprocess speech data for the first time with a VAE with two auxiliary variables and then uses its low dimensional representations as input to three downstream applications: speaker identification, gender classification, and emotion recognition. It is also the first time to use emotion recognition as a downstream application of a VAE.

The design of the model has the purpose of disentangling between these two kinds of long-term characteristics and also the instantaneous content of information using an auxiliary variable for each. After deriving the mathematical model for a VAE with two auxiliary variables starting with the standard evidence lower bound (ELBO) used for VAEs, results on testing the model will be displayed.

2 Related Work

Over the last years, many architectures were introduced that aim to extract low-dimensional useful representations from high-dimensional data such as images and audio data. An architecture based on variational autoencoders was developed to process audio data [2]. It used only one latent variable and was trained on a small dataset consisting of only 123 utterances in Spanish. The model did deliver slightly better results than the RBM (Restricted Boltzmann Machine), but not good enough for a downstream task.

Later, a model was developed to distinguish global from local timescale speech characteristics in audio data [12]. Its design is based on extending the variational autoencoder with an auxiliary variable h for capturing the global timescale features. This method created an architecture with four probability density distributions. The new architecture consists of the following subnetworks:

1. Global timescale network: takes as input the preprocessed speech audio dataand outputs a global latent variable h, which represents information extracted from an entire input-sequence.
2. Local time-scale network: takes as input a concatenation between the sameinput to the global time-scale network and its output. Its output is the local latent variable z. These two networks form unitedly the encoder.
3. The decoder network: takes the local latent variable to predict the nextspeech frame or make a reconstruction on the same input.
4. The predictor network: processes the local latent variable z to extract information from the global time-scale latent variable h hidden inside the local variable z. Its output is the global timescale variable h.

The model was trained and tested on the LibriSpeech dataset [7] (see section 4.1). The performance of this model is compared with the performance of other

models in section 5.1. The model has shown excellent results of recognizing the gender of the speaker as well as making speaker identification.

A byte-level language model based on recurrent neural networks (RNNs) was introduced to perform sentiment analysis [11]. Radford et al. (2017) aimed to use a simple recurrent neural network (1 layer with 4096 neurons) to get a sufficiently high-level representation of language data with disentangled features. Amazon reviews were used to train and test the model. The learning process was achieved by feeding the network one character (1 byte) and asking it to predict the next one. That means that the new representation of the input is not the output of the network. It is, however, the hidden states. The model has shown good results on sentiment analysis but was not tested on other downstream applications that we are interested in, such as speaker identification and gender classification.

A predictive model was developed for the same goal of learning useful representations from high-dimensional data like images and speech [6]. It was called Contrastive Predictive Coding (CPC), which uses a probabilistic contrastive loss function to finetune the network parameters. For audio speech data, the model was tested on the same LibriSpeech dataset for speaker identification.

3 Proposed Model

Deriving the Mathematical Model A variational autoencoder solves the problem of discontinuity in the latent space of autoencoders with variational inference, which results in a continuous latent space. Each of the two networks (the encoder and the decoder) is represented as a statistical model in Bayes formula:

$$p(z|x) = \frac{p(x|z) \cdot p(z)}{p(x)}$$

where $p(x)$ is the likelihood of generating the data x, $p(z|x)$ represents the encoder and $p(x|z)$ represents the decoder. Using the latent variable z, we could define a probabilistic model that could learn the distribution of the data x by defining the likelihood $p(x)$, using the tractable components $p(x|z)$ and $p(z)$, as:

$$p(x) = \int p(x, z)dz$$

In this equation we could add another component $\frac{p(z)}{p(z)}$ which does not change it, assuming that $p(z) \neq 0$:

$$p(x) = \int \frac{p(x, z)}{p(z)} \cdot p(z)dz$$

Now we apply logarithm on both sides, and then Jensens inequality rule on the logarithm and the integral on the second side of the equation to set a lower bound on $log[p(x)]$:

$$log[p(x)] \geq E_{p(z)}[log[\frac{p(x, z)}{p(z)}]]$$

$p(z)$ is the prior of z which is a wide distribution with vast areas that are not important for the data distribution, so we replace it by the posterior $p(z|x)$, but calculating the posterior requires components that we even need for our equation, so we approximate it with a function $q(z|x)$. This function can be calculated with a neural network. Now we need to add another term to the equation to ensure that $q(z|x)$ will approximate $p(z|x)$ with a tiny error. For that, we can use the Kullback Leibler divergence (KL). The KL divergence determines how different two distributions are from each other, and it is given for $q(x)$ and $p(x)$ as $KL(q(x) \mid\mid p(x)) = \int q(x) \cdot log[\frac{q(x)}{p(x)}]\ dx$, by adding this term to the last equation:

$$\begin{aligned} log[p(x)] &\geq E_{q(z|x)}[log[\frac{p(x,z)}{q(z|x)}]] + KL[q(z|x)\ ||\ p(z|x)] \\ &\geq E_{q(z|x)}[log[p(x,z)]] + H[q(z|x)] + KL[q(z|x)\ ||\ p(z|x)] \end{aligned}$$

where $H[q(z|x)]$ is the entropy given by: $H[q(z|x)] = -\int q(z|x) \cdot log[q(z|x)]\ dz$. In a similar approach, we consider now the existence of another two sources of information, h and m, which leads to:

$$p(x) = \int\int\int p(x,h,m,z)\ dh\ dm\ dz$$

Now we would multiply and divide with the joint probability of the latent variables $p(h,m,z)$. This probability represents the prior probability which means, that there are many values that do not matter for the desired data distribution x. So we take the posterior instead of the prior probability as it was done before when we took $p(z|x)$ instead of $p(z)$. As for $p(z|x)$, $p(h,m,z|x)$ is impossible to calculate, so we take a similar function $q(h,m,z|x)$ generated by a neural network. The KL divergence term will be ignored since it is not very important for the cost function. That leaves us with:

$$\begin{aligned} p(x) &= \int\int\int p(x,h,m,z) \cdot \frac{q(h,m,z|x)}{q(h,m,z|x)}\ dh\ dm\ dz \\ &= E_{q(h,m,z|x)}\ [\frac{p(x,h,m,z)}{q(h,m,z|x)}] \end{aligned}$$

Now we take the logarithm of both sides as before and apply the Jensen rule after that:

$$\begin{aligned} log[p(x)] &\geq E_{q(h,m,z|x)} log[\frac{p(x,h,m,z)}{q(h,m,z|x)}] \\ &\geq E_{q(h,m,z|x)} log[p(m|h,z,x)] + \\ &\quad E_{q(h,m,z|x)} log[p(h|z,x)] + E_{q(h,m,z|x)} log[p(x|z)] + \\ &\quad E_{q(h,m,z|x)} log[p(z)] - E_{q(h,m,z|x)} log[q(z|h,m,x)] - \\ &\quad E_{q(h,m,z|x)} log[q(m|h,x)] - E_{q(h,m,z|x)} log[q(h|x)] \end{aligned} \tag{1}$$

$$
\begin{aligned}
-E_{q(h,m,z|x)} log[q(z|h,m,x)] &= -\int q(h,m,z|x) \cdot log[q(z|h,m,x)] \\
&= -\int q(z|h,m,x) \cdot q(h,m|x) \cdot log[q(z|h,m,x)] \\
&= E_{q(h,m|x)} H[q(z|h,m,x)]
\end{aligned} \tag{2}
$$

$$
\begin{aligned}
-E_{q(h,m,z|x)} log[q(m|h,x)] &= -\int q(h,m|x) \cdot log[q(m|h,x)] \\
&= -\int q(m|h,x) \cdot q(h|x) \cdot log[q(m|h,x)] \\
&= E_{q(h|x)} H[q(m|h,x)]
\end{aligned} \tag{3}
$$

$$
-E_{q(h,m,z|x)} log[q(h|x)] = -E_{q(h|x)} log[q(h|x)] = H[q(h|x)] \tag{4}
$$

By replacing (2), (3) and (4) in (1) and removing variables that do not have an effect on the distribution and introducing hyperparameters α, β, and γ, we get:

$$
\begin{aligned}
log[p(x)] \geq\ & E_{q(z|x)} log[p(x|z)] + \\
& \alpha(\ E_{q(h|x)} log[p(z)] + E_{q(h,m|x)} H[q(z|h,m,x)]\ ) + \\
& \beta(\ E_{q(h,z|x)} log[p(m|h,z,x)] + E_{q(h|x)} H[q(m|h,x)]\ ) + \\
& \gamma(\ E_{q(z|x)} log[p(h|z,x)] + H[q(h|x)]\ )
\end{aligned} \tag{5}
$$

This is the final equation for the cost function of a variational autoencoder with two auxiliary variables. The three hyperparameters work as regularization terms on the KL divergence [12].

4 Proposed Architecture

The model introduced in this work is based on the equation of the variational lower bound of an Aux-VAE with two auxiliary variables. There are six different probability distributions, each of which can be modeled using neural networks as shown in figure 1. These distributions are as follow:

1. $P(h|x)$ represents the global timescale network, which is modeled as follows: Three blocks, each consists of a convolutional layer with kernel size three and Tanh as an activation function, as well as a batch normalization layer. Followed by a convolutional layer with kernel size one. The stride value is one for all convolutional layers. A global average pooling layer GAP follows to encode global features from the input. The output of GAP is then turned into a normal distribution h with a mean and a standard deviation.

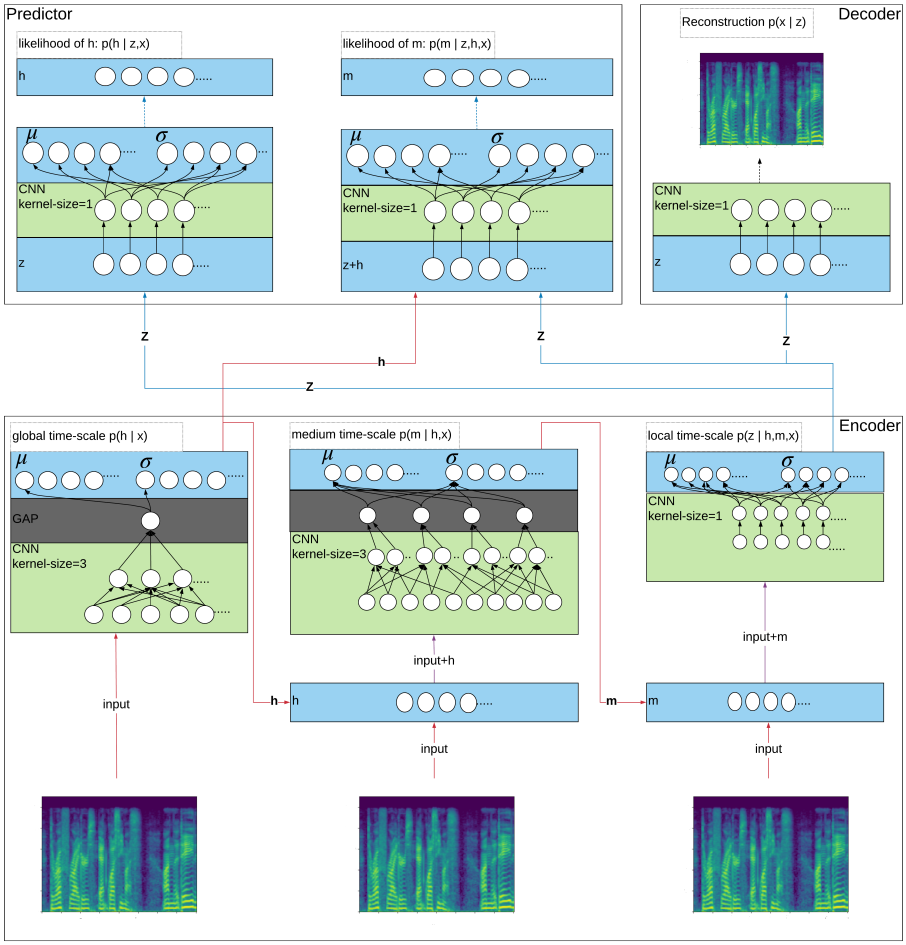

Fig. 1: The structure of the network. The same input goes into the three subnetworks of the encoder. The output of the decoder is the same as its input. Over each subnetwork, the distribution which it represents is written as label.

2. $P(m|h,x)$: represents the medium time-scale network, which has the same first three blocks as $p(h|x)$ and then a convolutional layer with kernel size one. After that, instead of applying only one global average pooling on the whole output, the average pooling is applied on a medium duration (1 sec) of the output. That makes it possible to encode features in a way different from both the global and the local time-scale networks, an encoding between both. The output of the GAP is turned into a normal distribution called m. This network takes as input a concatenation between the same input as before and a random sample from the distribution h.
3. $P(z|h,m,x)$: represents the local time-scale network. It is modeled by a CNN, which consists of three convolutional layers with kernel size one and batch normalization layers. The Output of the CNN is turned into a normal

distribution z. This network takes as input a concatenation between the same input to the last two networks and a random sample from the distribution m.

4. $P(x|z)$: represents the reconstruction network for the input x using the latent variable z. It is also modeled by a CNN with four convolutional layers with kernel size one and batch normalization layers. It takes as input a random sample from the distribution z and reconstructs x from it.
5. $P(m|h, z, x)$: represents the reconstruction network for m using h, z, and x, which is implemented inside z. This network is a CNN of three convolutional layers with kernel size one and batch normalization layers. It takes as input a concatenation between a random sample from h and a random sample from z to output the distribution m.
6. $P(h|z, x)$: represents the reconstruction network for h using z and x, which is implemented inside z. This network is a CNN of two convolutional layers with kernel size one and batch normalization layers. It takes as input a random sample from z and outputs the distribution h.

The first three networks model the encoder, while the forth network models the decoder. The later two networks model the predictor, which predicts the globaland medium-timescale variables from the lcoal-timescale varible.

5 Datasets and Training

5.1 Datasets

LibriSpeech is a corpus of read English speech containing about 1000 hours of speech derived from audiobooks and sampled at a rate of 16kHz. It comprises many subsets such as train-clean-100, which was used in this work for speaker identification and gender recognition. This subset contains 100 hours of audio data taken from 251 speakers (126 males and 125 females). The dataset was used very often to train and test models on speaker identification, as in [12] and [6].

One-Minute-Gradual (OMG) Emotion Dataset is a dataset consisting of about 8 hours of emotional monolog youtube videos. The videos were separated based on utterances and individual utterances were labeled separately by humans. The labels are the arousal and the valence of the utterance as well as the overall utterance-level emotion. The used emotions are anger, disgust, fear, happy, neutral, sad, and surprise [1].

5.2 Preprocessing and Training

Preprocessing The main goal here is to reach an effective spectrogram from the audios to be used as input for the variational autoencoder. We found experimentally that a Mel-spectrogram with 140 frequencies achieves these requirements.

Training The model was trained first for 40 epochs on the LibriSpeech dataset for speaker identification, and after that, 40 epochs on the OMG-Emotion dataset to perform emotion regression with the following hyperparameters for both training processes: Learning rate = 0.0001, batch size = 32 and adam as an optimizer. The hyperparameters of the loss function α, β and γ were determined empirically and set to 0.01.

6 Experiments

6.1 Speaker Identification and Gender Classification

Speaker identification is basically a classification task, where the model takes an audio speech record as input and outputs the label of the speaker. Recently, it was often used as a downstream task to assess how powerful unsupervised-learned representations are. Since the goal of this work is proving the effectiveness of the learned representations by the multi-timescale Aux-VAE proposed in this work, speaker identification and gender classification are essential tasks to compare among the results of the three learned representations by the model, and also with results from previous works. The first step is training the VAE to learn how to extract the representations and then using them as input to a linear classifier. The same procedure was made and the same LibriSpeech dataset was used by [12] for training the Aux-VAE and by [6] for training the CPC.

Table 1: Speaker ID accuracy results compared with previous works on the LibriSpeech dataset; h denotes the global variable, m denotes the medium variable, and z denotes the local variable.

Model	Accuracy
Aux-VAE; h [12]	95.3
Aux-VAE; z [12]	98.1
CPC [6]	97.4
Supervised [6]	98.5
VAE with two auxiliaries; h	**99.86 ± 0.05**
VAE with two auxiliaries; m	98.30 ± 0.20
VAE with two auxiliaries; z	99.68 ± 0.08

As we see in table 1, the accuracy results of our multi-timescale Aux-VAE using each the global and the local time-scale variables have outperformed the results of both the AuxVAE and the CPC and also the supervised model that was mentioned in [6]. The accuracy results on gender classification are also excellent: global=97.66%, medium=96.34%, and local=98.14%. We can see as well that the global time-scale variable has outperformed the medium variable on both classification tasks, which supports the fact that the global latent variable learns features that are related to the speaker identity better than the medium variable. The local timescale variable was also better than the medium, which was expected because the local latent variable takes as a part of its input a

sample from the medium variable, which in his turn takes as a part of its input a sample from the global variable. With that, the local latent variable was able to capture some features by its composed input that the medium variable has missed.

6.2 Visualizing the latent spaces using T-distributed Stochastic Neighbor Embedding (t-SNE)

T-SNE is an algorithm developed for visualizing multidimensional data, based on the idea of dimensionality reduction. We will use t-SNE plots as in [12] to visualize the latent variables of the VAE and to get an optical feeling of how good they perform. First, the latent representations were acquired for each audio speech record of 8 different speakers with equal gender separation (four females and four males). Each audio record is now a multidimensional data point (512 dimensions). T-SNE was then used to reduce the dimensions to only two for a 2D plot. The speakers from the LibriSpeech dataset were also used in [12] for visualizing the global variable of the Aux-VAE. The clustering and separation between different speakers and genders is here clearer than in [12].

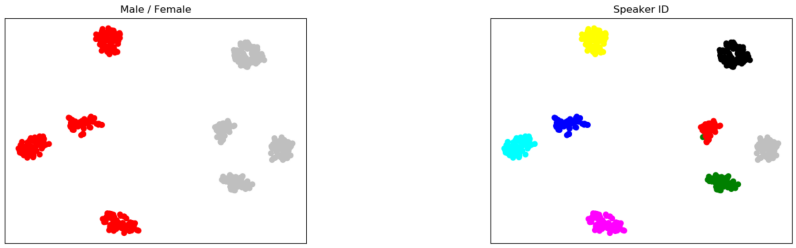

Fig. 2: A plot from the global latent space; the left figure shows the separation between male and female; the right one shows different speakers as the data from each speaker come together to form a separable cluster.

Visualizing the global time-scale variable In figure 2, we can see the perfect separation between the two genders, where only drawing one line would be enough to separate the classes. We also see a perfect separation among the speakers; data points of each speaker has made their own cluster.

Visualizing the medium time-scale variable shows that the separation between the two genders is not as perfect as in figure 2, where a single line would have been enough to separate the two classes. The visualization also shows a good separation between the speakers. Such results are expected since the results of the speaker identification in section 5.1 were also excellent.

6.3 Emotion Recognition

Each emotion or affective experience can be expressed using at least two properties: arousal, which is a measurement between pleasant and unpleasant, and valence, which is a measurement between quiet and active [5]. So instead of fitting a model to classify different emotions such as happy, angry, and sad, we have a downstream application for predicting these two values using a regression model. As in section 5.1 the learned representations of a trained multi-timescale Aux-VAE were used as input to this downstream application. The model of this application has only two dense-layers, one hidden layer and an output layer.

The different results from using different latent variables were compared among each other and among results of previous works published in the OMG-Emotion challenge[1]. The measure of comparison is the same as in the challenge, the Concordance correlation coefficient (CCC), which measures the correlation between two variables (larger is better). These two variables are here the annotations and the output of the model (what we expect the model to give as output and its real output). Since choosing the right loss function gives better results, two different losses were used for training to see which one delivers better results on CCC. First, the model was trained to minimize the mean square error (MSE) between its output and the labels and then using the resulting model to calculate CCC values. Second, the model was trained to maximize CCC directly. From table 2, we can notice that the CCC value got better on all latent variables and for both arousal and valence when using the CCC as a loss function. Also, we notice instantly that the medium variable is the best among the three in terms of predicting the value of both arousal and valence.

Table 2: CCC values of arousal and valence when using the different latent variables. In the first column, the downstream model was trained with MSE. In the second column, the downstream model was trained with CCC; h denotes the global variable, m denotes the medium variable, and z denotes the local one.

Property; Variable	MSE Training	CCC Training
Arousal; h	0.14668	0.33457
Arousal; m	**0.30027**	**0.3531**
Arousal; z	0.26032	0.32033
Valence; h	0.30452	0.34355
Valence; m	**0.36961**	**0.37399**
Valence; z	0.30650	0.36010

We see also that the CCC values obtained from the medium variable are better than the baseline and all other models that are based on audio for training and testing (see table 3).

[1] `https://www2.informatik.uni-hamburg.de/wtm/omgchallenges/omg_emotion2018_results2018.html`

Table 3: Compares the CCC value on arousal and valence of previous works with m.

Model	Arousal	Valence
AudEERING [13]	0.2925419226	0.361123256087
ExCouple [9]	0.182249525233	0.211179832035
Peng et al. [8]	0.1879	0.256
Deng et al. [3]	0.273	0.266
Pereira et al. [10]	0.17	0.16
Baseline [1]	0.15	0.21
Medium variable m	**0.36961**	**0.37399**

6.4 Gender and Emotion Transformation

These tasks were performed by feeding the global variable (for gender transformation) or the medium variable (for emotion transformation) different input from the others so that the global or the medium encodes the high-level characteristics of a speaker (from opposite gender or other emotion) and give them to the local variable. It can then build a new z distribution for the reconstruction process on the input x. As an example, a gender transformation from male to female was performed. That means the global variable took as input speech data from a female, while the local and medium variables took speech data from a male. The results can be heard in the git repository[2] under "experiments/reconstructions". Gender transformation was performed before using an Aux-VAE with one auxiliary variable [12]. In figure 3, we can notice that the content of the audio is still the same, but the fundamental frequency was raised a little bit with some changes.

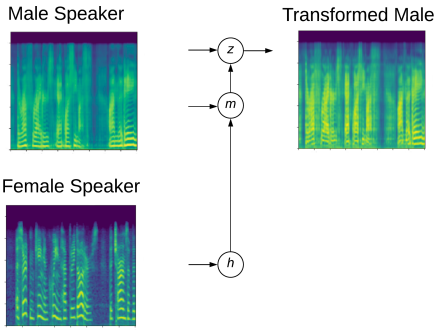

Fig. 3: Transforming the Mel spectrogram of a male using the global characterstics of a female speaker.

Content of the audio is still the same, but the fundamental frequency was raised a little bit with some changes.

[2] `https://github.com/Hussam-Almotlak/voice_analysis`

7 Conclusion

We have developed an Aux-VAE, which includes beside the global variable h also an additional medium variable m. We tested the model on audio speech data, where the architecture worked as a multi-timescale model. The experiments have shown that the global variable h is better than the medium m on speaker identification and gender classification, while m is better than h on emotion regression. Our model has also exceeded the state-of-the-art on speaker identification, and emotion regression from speech.

References

1. Barros, P., Churamani, N., Lakomkin, E., Sequeira, H., Sutherland, A.,Wermter, S.: The OMG-Emotion Behavior Dataset. In: 2018 International Joint Conference on Neural Networks (IJCNN). pp. 1408–1414. IEEE (2018). https://doi.org/10.1109/IJCNN.2018.8489099
2. Blaauw, M., Bonada, J.: Modeling and transforming speech using variational autoencoders. In: INTERSPEECH. pp. 1770–1774 (2016)
3. Deng, D., Zhou, Y., Pi, J., Shi, B.E.: Multimodal utterance-level affect analysis using visual, audio and text features. arXiv preprint arXiv:1805.00625 (2018)
4. Hsu, W.N., Zhang, Y., Glass, J.: Learning latent representations for speech generation and transformation. arXiv preprint arXiv:1704.04222 (2017)
5. Kuppens, P., Tuerlinckx, F., Russell, J.A., Barrett, L.F.: The relation between valence and arousal in subjective experience. Psychological Bulletin 139(4), 917 (2013)
6. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
7. Panayotov, V., Chen, G., Povey, D., Khudanpur, S.: LibriSpeech: an ASR corpus-based on public domain audio books. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 5206–5210. IEEE (2015)
8. Peng, S., Zhang, L., Ban, Y., Fang, M., Winkler, S.: A deep network for arousal-valence emotion prediction with acoustic-visual cues. arXiv preprint arXiv:1805.00638 (2018)
9. Pereira, I., Santos, D.: OMG emotion challenge - ExCouple team. arXiv preprint-arXiv:1805.01576 (2018)
10. Pereira, I., Santos, D., Maciel, A., Barros, P.: Semi-supervised model for emotion recognition in speech. In: International Conference on Artificial Neural Networks. pp. 791–800. Springer (2018)
11. Radford, A., Jozefowicz, R., Sutskever, I.: Learning to generate reviews and discovering sentiment. arXiv preprint arXiv:1704.01444 (2017)
12. Springenberg, S., Lakomkin, E., Weber, C., Wermter, S.: Predictive auxiliary variational autoencoder for representation learning of global speech characteristics. Proc. INTERSPEECH 2019 pp. 934–938 (2019)
13. Triantafyllopoulos, A., Sagha, H., Eyben, F., Schuller, B.: audEERING's approach to the One-Minute-Gradual emotion challenge. arXiv preprint arXiv:1805.01222 (2018)