

SLR Report

Publications worth reading about pre-trained speech models:

1.) "DAWN OF THE TRANSFORMER ERA IN SPEECH EMOTION RECOGNITION: CLOSING THE VALENCE GAP"

<https://arxiv.org/pdf/2203.07378.pdf>

2.) "A Fine-tuned Wav2vec2.0/HuBERT Benchmark For Speech Emotion Recognition, Speaker Verification and Spoken Language Understanding"

<https://arxiv.org/pdf/2111.02735.pdf>

3.) "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations"

<https://arxiv.org/pdf/2006.11477.pdf>

4.) "HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units"

<https://arxiv.org/pdf/2106.07447.pdf>

5.) "UNSUPERVISED CROSS-LINGUAL REPRESENTATION LEARNING FOR SPEECH RECOGNITION" (XSLR)

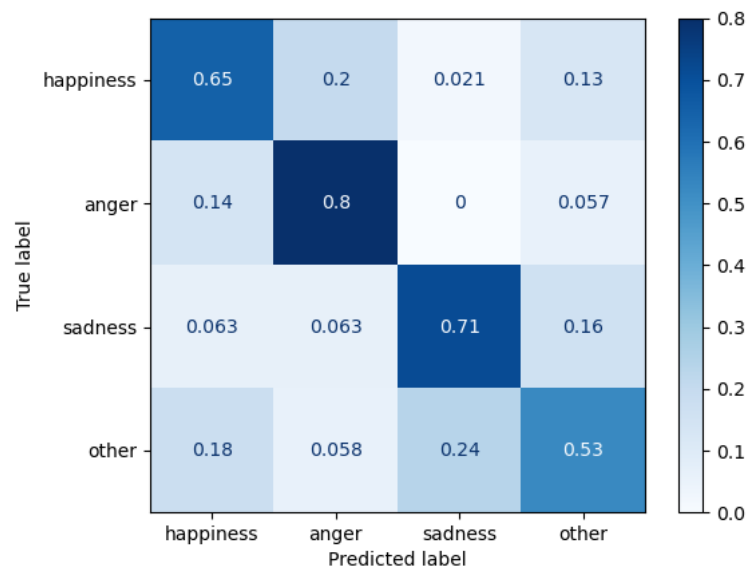
<https://arxiv.org/pdf/2006.13979.pdf>

Audeering model:

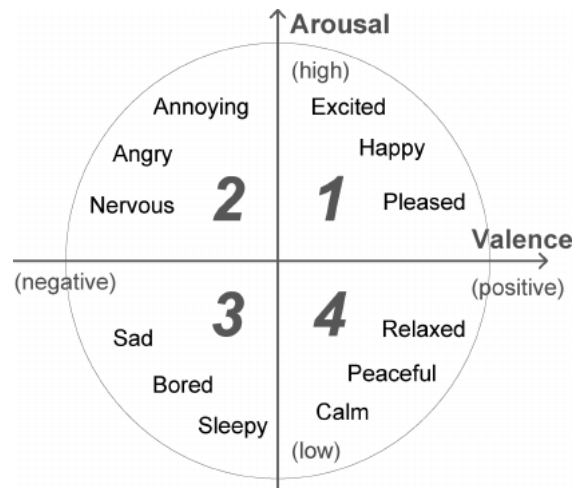
Results on the evaluation set from the embedding model (no fine-tuning on our data):

<https://huggingface.co/audeering/wav2vec2-large-robust-12-ft-emotion-msp-dim>

eval acc = 62.4%



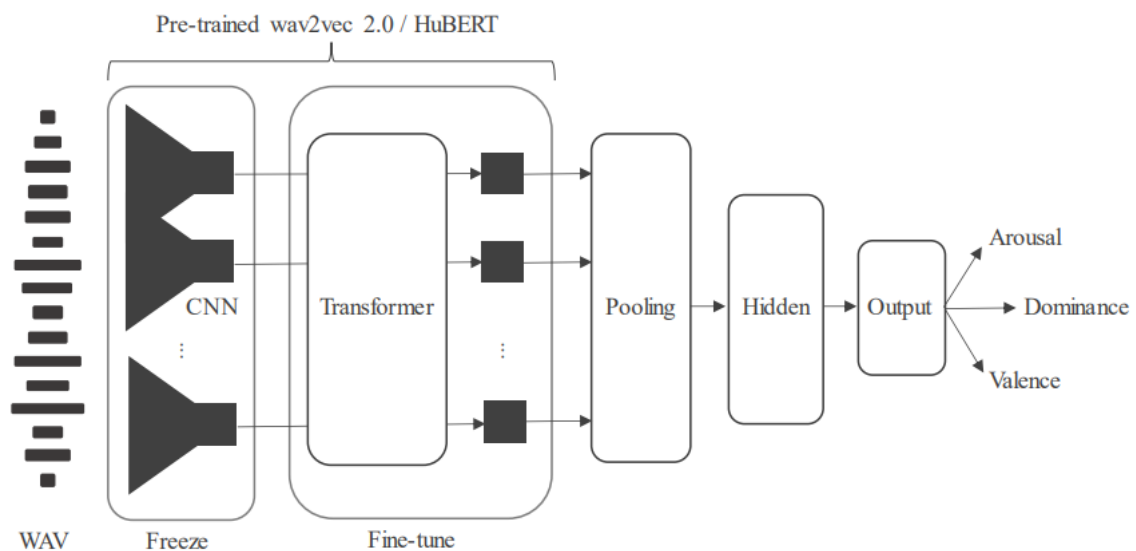
The original prediction comes in 3 classes in the range 0..1: valence, dominance, arousal



"For arousal and dominance, all tested models perform equally well, whereas with respect to valence / sentiment the data used for pre-training has a strong effect."

Publication describing this model:

<https://arxiv.org/pdf/2203.07378.pdf>



Architecture:

- Built on top of wav2vec 2.0 / HuBERT
- Average pooling over last transformer layers hidden states → hidden layer → output layer
- Parameters
 - Adam optimizer,
 - CCC loss (Concordance correlation coefficient) (for continuous values of arousal, valence and dominance)
 - $lr = 1e-4$
 - `batch_size = 32`
 - 5 epochs
 - Partial fine-tuning - freeze the CNN layers but fine-tune the transformer layers.

Dataset: MSP-Podcast, A large naturalistic speech emotional dataset, ≈100h

Publication: https://ecs.utdallas.edu/research/researchlabs/msp-lab/publications/Lotfian_2019_3.pdf

Description: <https://ecs.utdallas.edu/research/researchlabs/msp-lab/MSP-Podcast.html>

Note:

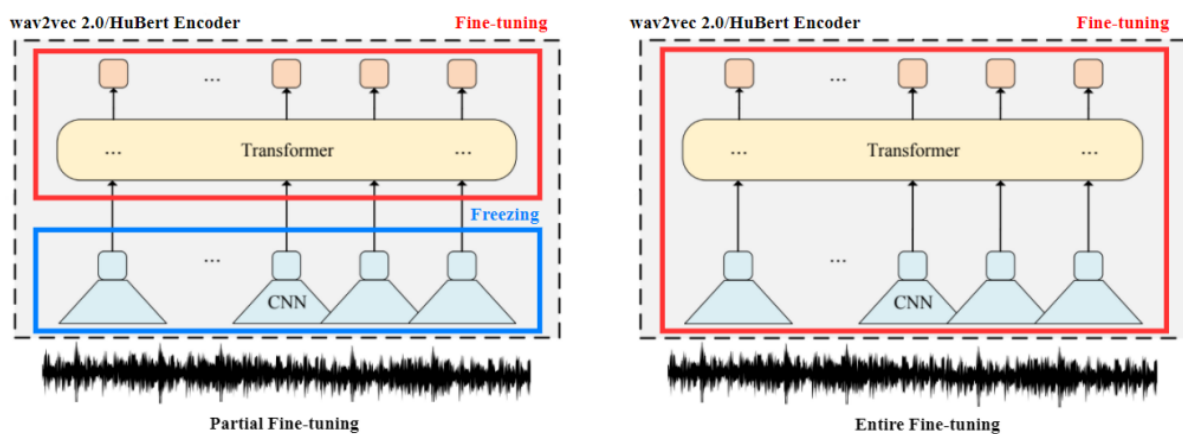
Only a few studies have evaluated performance on augmented test data as well - previous SER models show robustness issues, particularly for background noise and reverb.

Publication looks at Wav2vec2.0/HuBERT fine-tuning methods for Speech Emotion Recognition (SER), Speaker Verification (SV) and Spoken Language Understanding (SLU)

<https://arxiv.org/pdf/2111.02735.pdf>

“partial fine-tuning appears to be a better fine-tuning method than entire fine-tuning for SER”

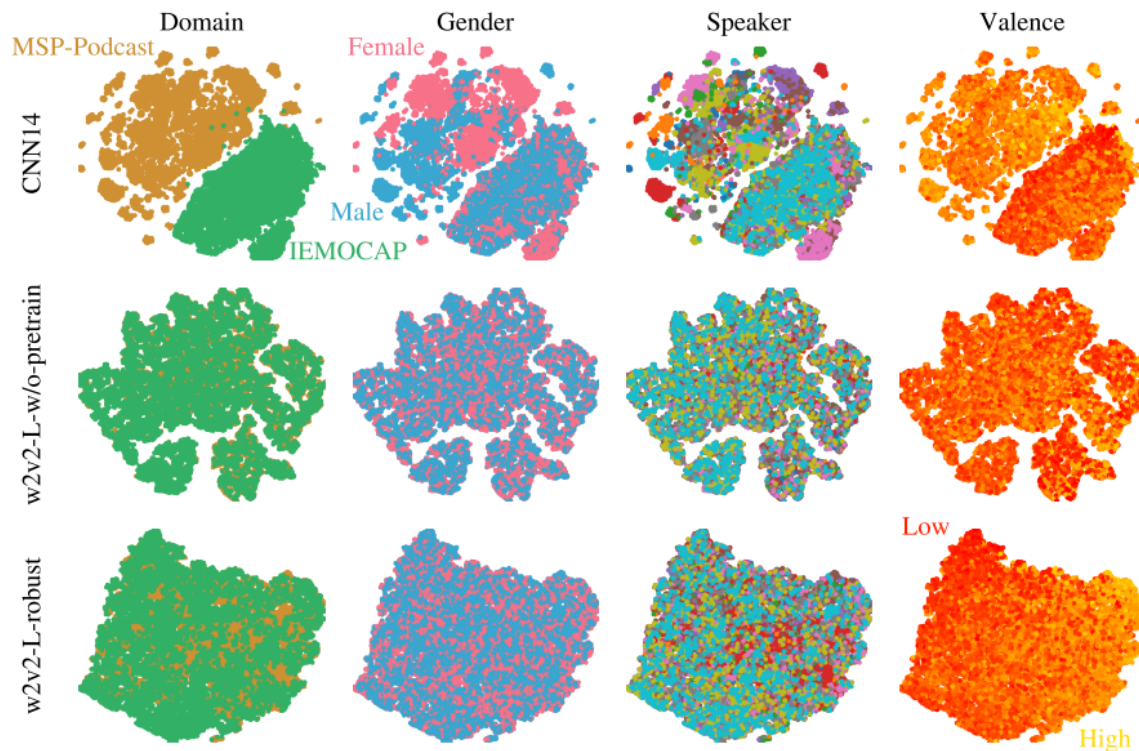
for SV entire fine-tuning outperforms partial fine-tuning - could use them for voiceid.



Tutorial on fine-tuning:

<https://huggingface.co/blog/fine-tune-xlsr-wav2vec2>

Different model type embeddings:



No clusters for domain, gender or speaker for both w2v2 models, but the pre-trained one (w2v2-L-robust) is the only one that shows a rather smooth transition from low to high valence scores.

Conclusions:

1.) A larger architecture does not lead to better performance per se. Larger architectures using different data during pre-training might perform worse than smaller architectures. (although a comparison was made between 95M vs 316M models, my previously used 20M model is still much smaller)

It was concluded that transformer layers can be reduced to 12 without a degradation in performance. With less than 12 layers we begin to see a negative effect on valence. That results in 164M params for the w2v2-L-robust from the original 316M.

2.) Most models show good sex fairness scores and sex mean shift values for arousal and dominance. For valence, most models show a higher CCC (Concordance correlation coefficient) for females than for males.

3.) Models pre-trained on multiple languages seem to benefit from added linguistic information (BERT embeddings were concatenated with the current transformer models embeddings and a regression head was trained, both model's weights were frozen)

The models that benefit most are the two multi-lingual models w2v2-L-vox and w2v2-L-xls-r, which gives some evidence that it is in particular models pre-trained on multiple languages that gain from a fusion with text features. (The base for the current model is the Wav2Vec2-Large-Robust)

4.) Even without pre-training, the latent space provided by the transformer architecture generalises better than CNN14, as it abstracts well away from domain and speaker. In case of valence, however, a pre-training is necessary, as otherwise, prediction fails.

(This could explain our models inability to differentiate between happiness and anger - those are mainly dependent on valence.)

5.) Fine-tuning of the transformer layers is necessary and worth the computational cost it incurs. (As opposed to freezing the whole model and just training the last fc layers). At the same time partial fine-tuning appears to be a better fine-tuning method than entire fine-tuning for speech emotion detection SER. (frozen CNN layers)

Experiments also have shown that models that see the biggest performance gain due to an adaptation of the self-attention layers are hubert-L and w2v2-L-robust - the same ones that benefitted the least from additional text information in the form of BERT embeddings.

6.) The models are able to implicitly capture linguistic information from audio only. To what extent they learn sentiment during fine-tuning, though, depends on the data used for pre-training (e. g. multi-lingual data makes it more difficult). Generally, we see that the performance on valence correlates with a model's ability to predict sentiment.

Sentence	CNN14	w2v2-L-robust	w2v2-L-xls-r
This is wonderful	.554	.816	.428
This is stupid	.561	.137	.449
Wonderful stupid	.545	.327	.502
In the afternoon	.416	.588	.571
Behind the wall	.534	.539	.425

1.) Happiness/anger predictions depend on the ability to determine valence score, which itself is highly correlated with the usage of linguistic information. w2v2-L-robust model is pre-trained and fine-tuned on so much English language data that it has learned the linguistic information by itself. Multi-lingual models benefit from that as well, by showing an increase in valence prediction capability after being given the embeddings from the BERT language model. (test set in English).

model type	training type	best test acc	train accuracy	eval accuracy	baseline eval acc	API acc	dataset
russian	classification	0.730	0.986	0.675	0.598	0.68	
russian	contrastive	0.757	0.903	0.660	-	0.63	
audeering	classification	0.882	0.923	0.597	-	0.661	
audeering	contrastive	0.725	0.946	0.662	0.624		

type	wav2vec type	languages pre-training	languages finetuning	transformer layers, param count	param count
russian	w2v2-L-XLRS	multi-lingual	russian (3.5h)	24, 316M	316M
audeering	w2v2-L-robust	english	english (85h)	12, 160M	160M

Wav2Vec2-XLS-R-300M - <https://huggingface.co/facebook/wav2vec2-xls-r-300m>

Wav2Vec2-Large-Robust - <https://huggingface.co/facebook/wav2vec2-large-robust>

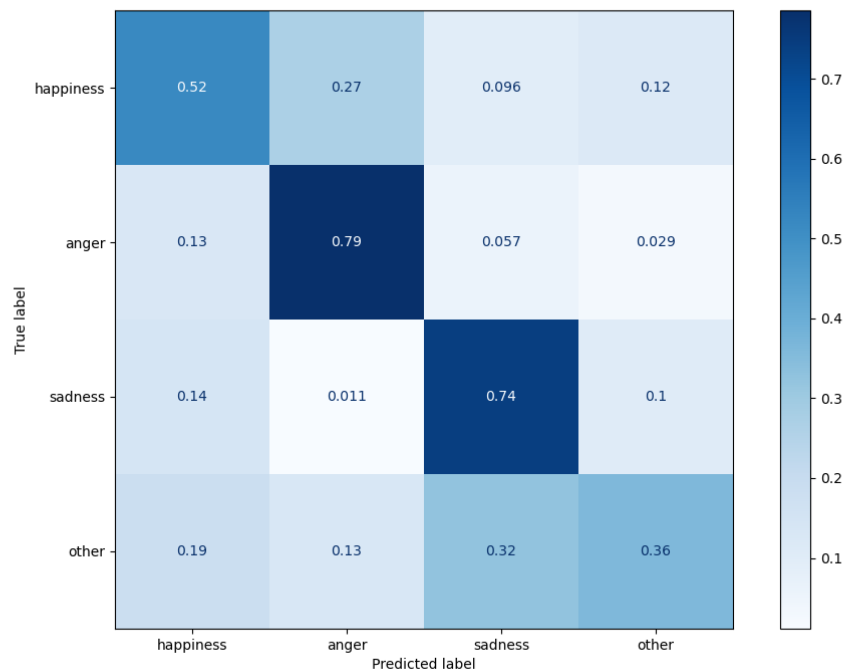
Results from fine-tuned Russian models:

The experiments here (1, 2 and 3) trained the whole model, including the CNN layers. Now considering the latest conclusions, a better fine-tuning method would have been to freeze the CNN layers too and reduce the learning rate.

1.) Classification based on Russian model (not frozen model weights)

train accuracy = 0.781, test acc = 0.692, eval acc = 0.533

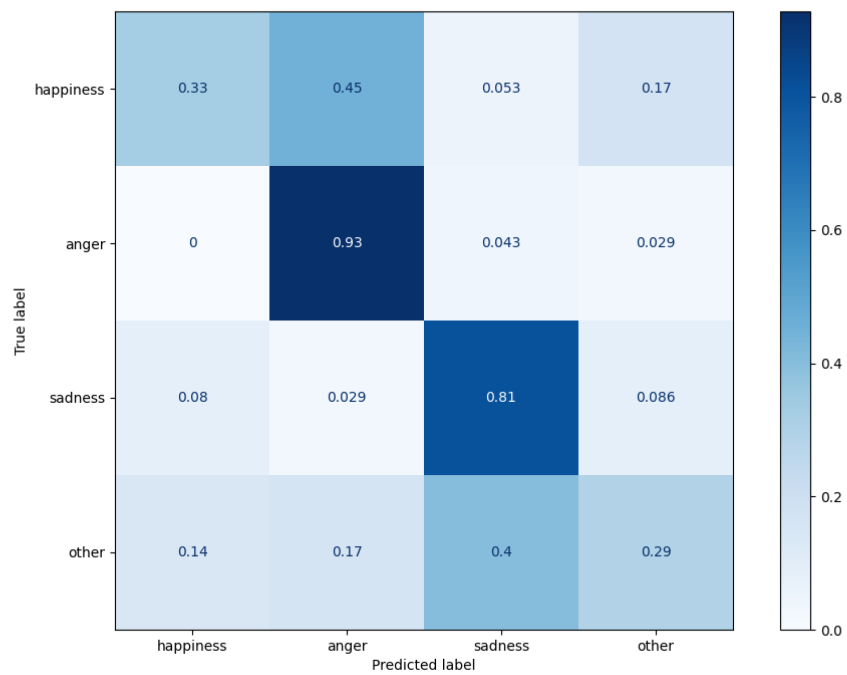
eval matrix:



2.) Classification based on Russian model (frozen model weights)

train accuracy = 0.774, test acc = 0.658, eval acc = 0.505

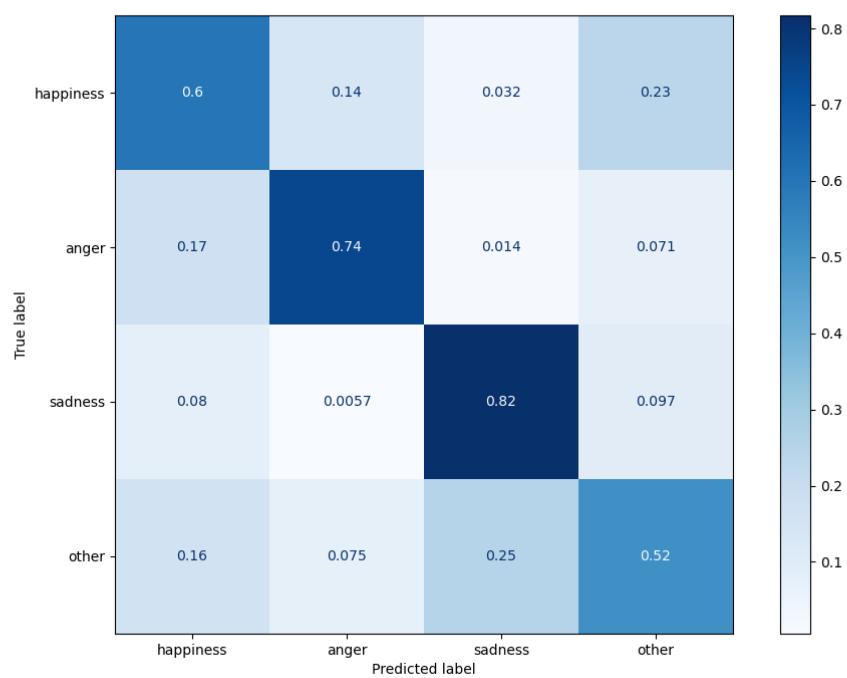
eval matrix:



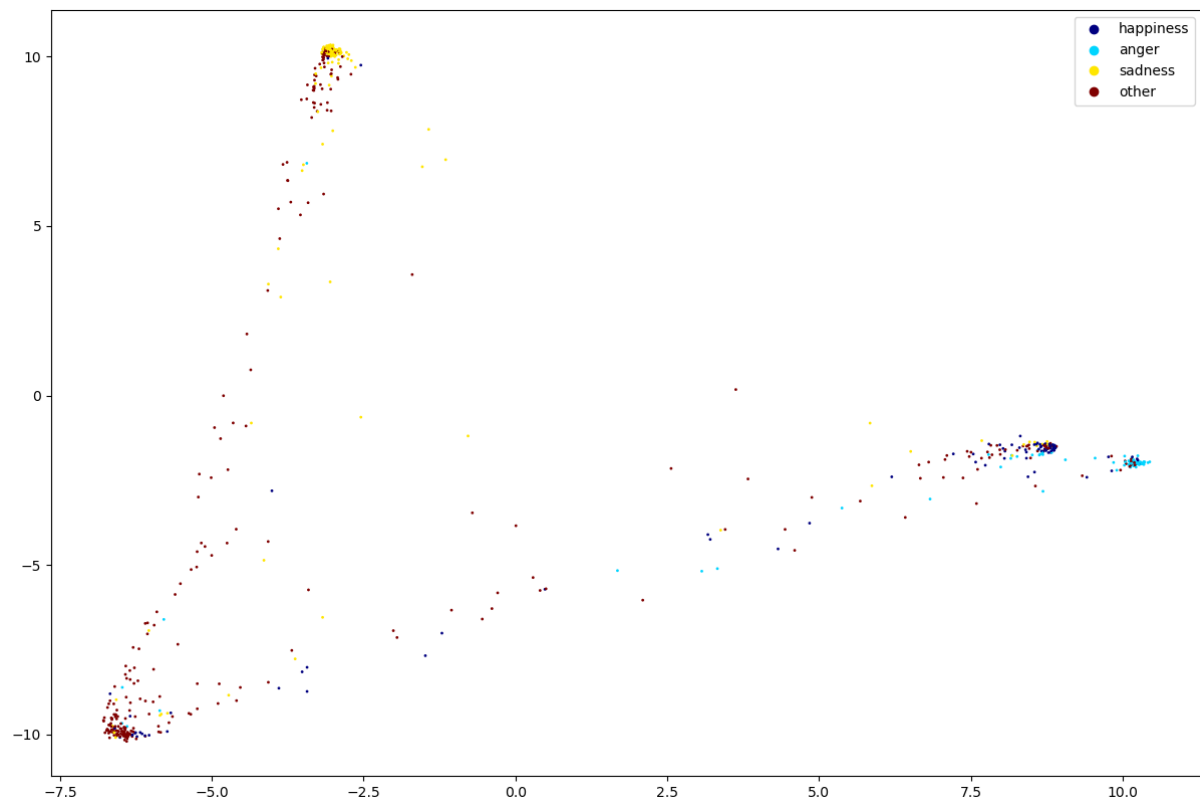
3.) Contrastive loss based on Russian model (not frozen model weights, including conv layer)

train accuracy = 0.98, test acc = 0.63, eval acc = 0.634

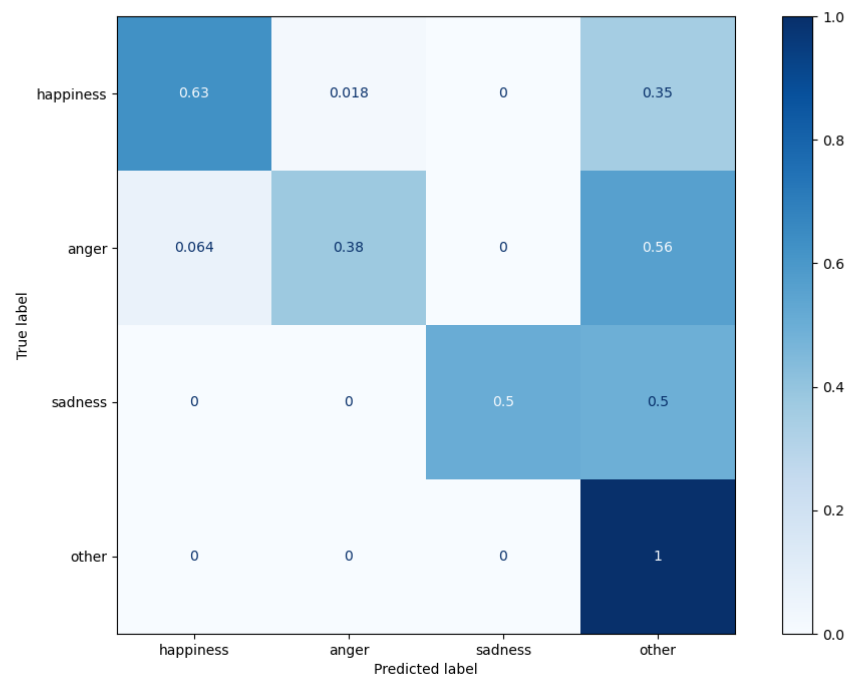
eval matrix:



eval embeddings:



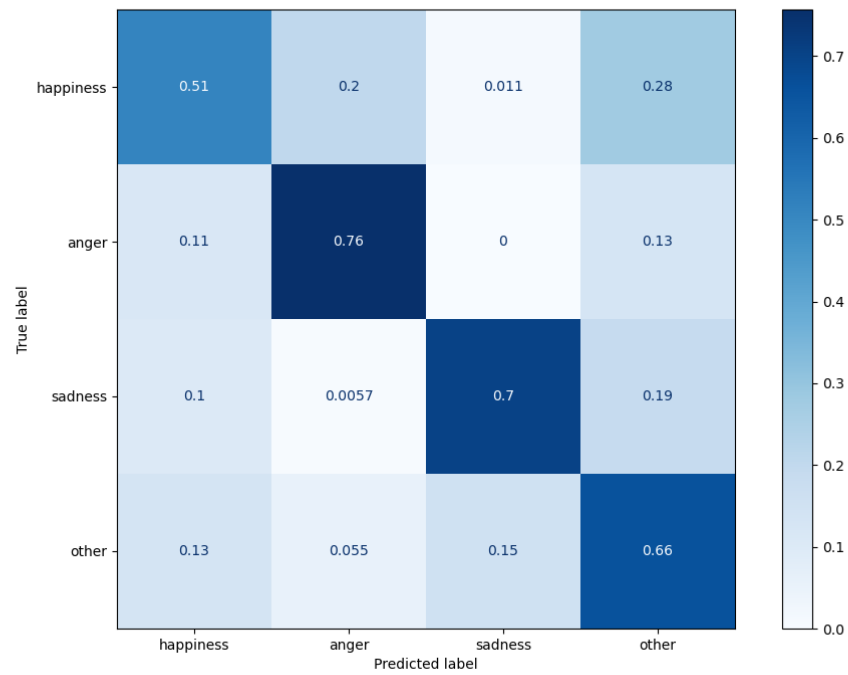
test matrix:



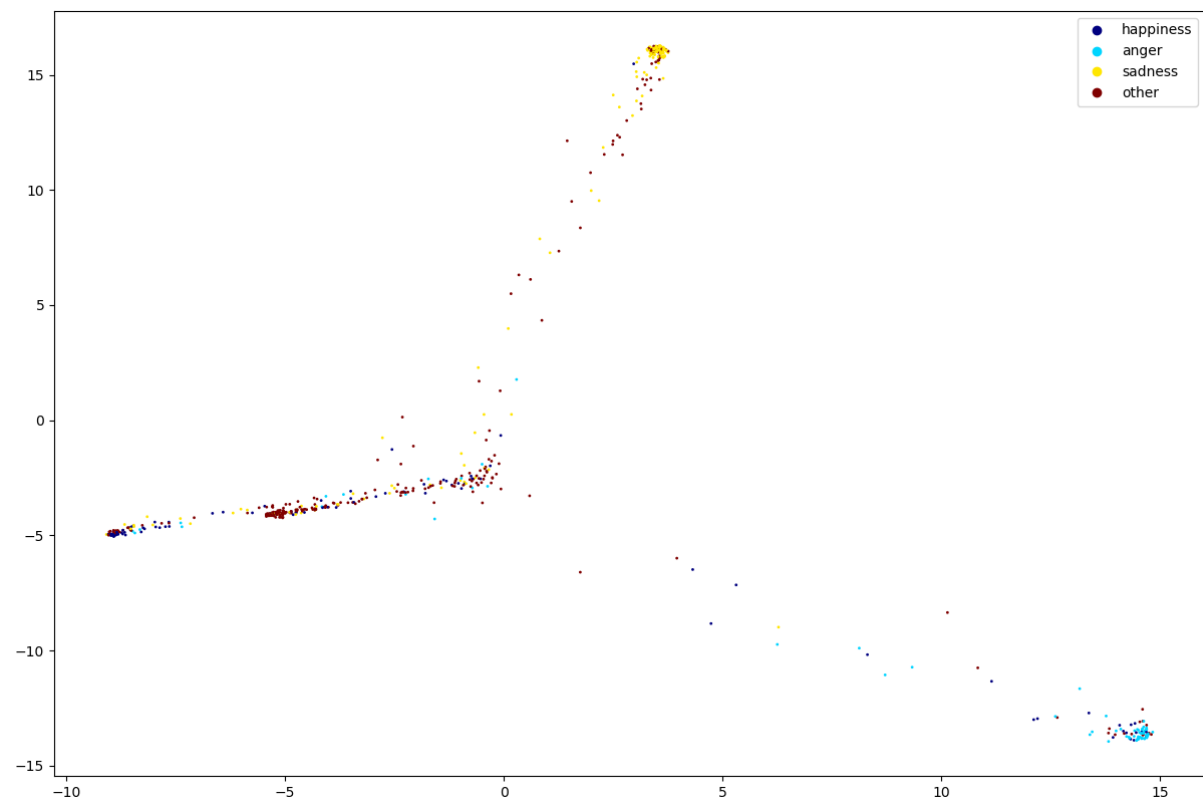
4.) Contrastive loss based on Russian model (not frozen model weights, conv layer frozen - as in publication, lr = 1e-5, dropout=0.5, batch_size = 8)

train accuracy = 0.903, test acc = 0.757, eval acc = 0.660

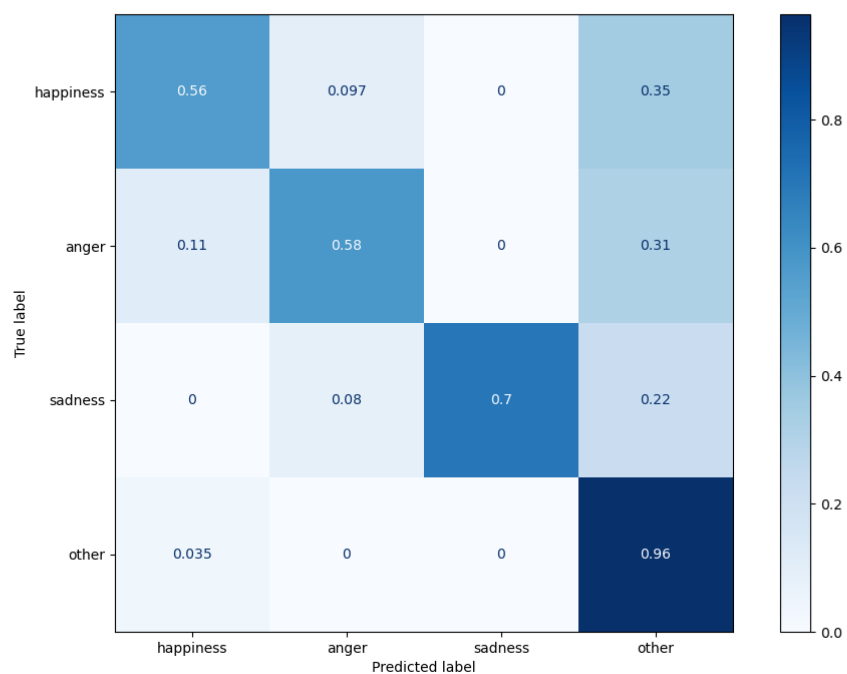
eval matrix:



eval embeddings:



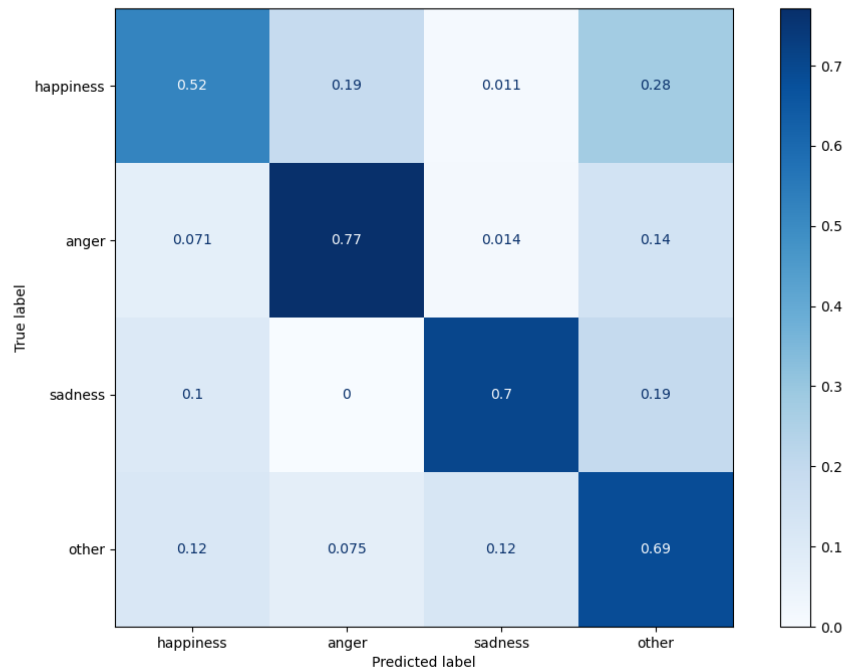
test matrix:



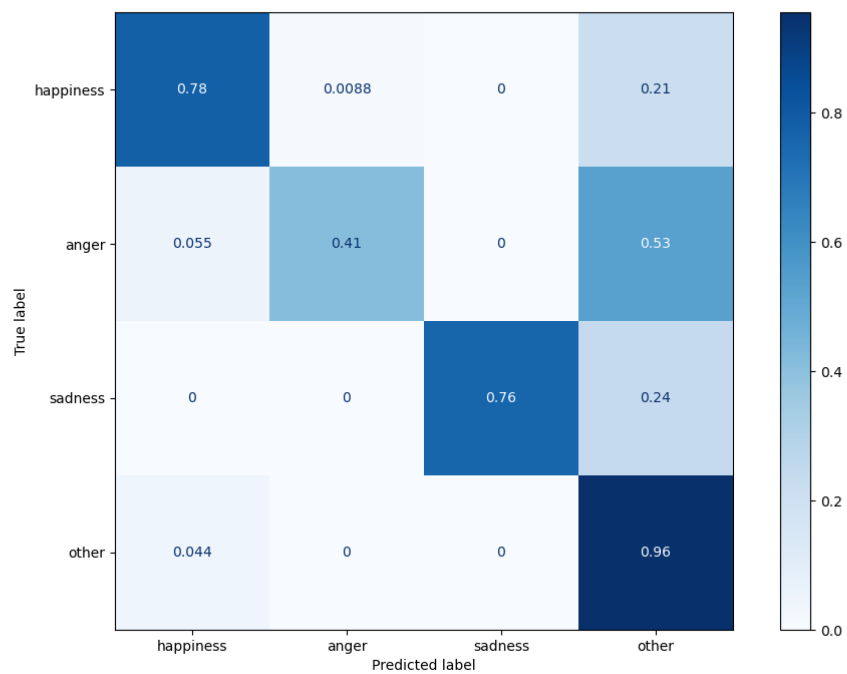
5.) newest classification, Russian model (frozen conv layers, lr = 1e-5, batch_size = 8, dropout=0.5)

train accuracy = 0.986, test acc = 0.730, eval acc_1 = 0.675, eval acc_2 = 0.82

eval matrix:



test matrix:

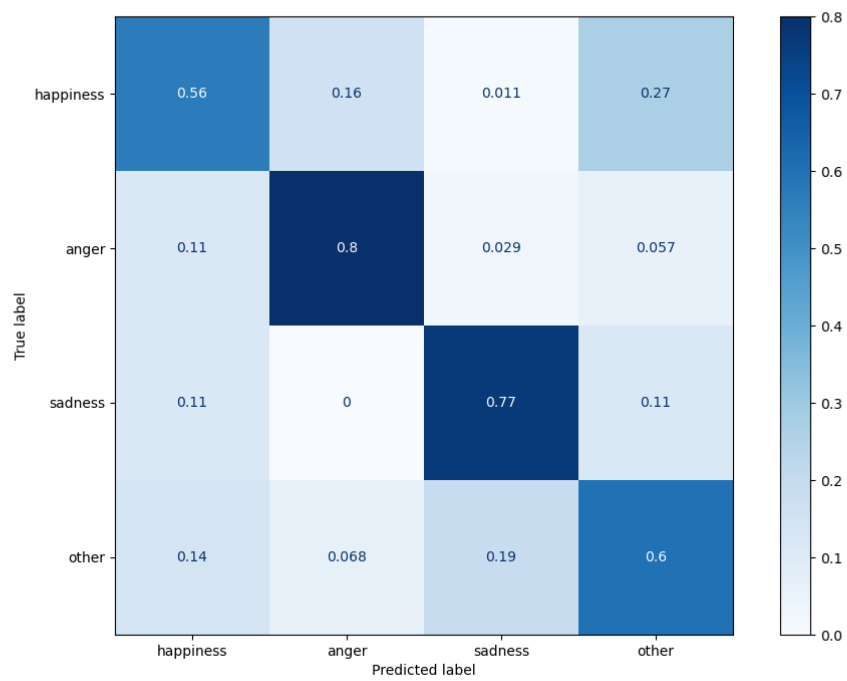


Audeering model

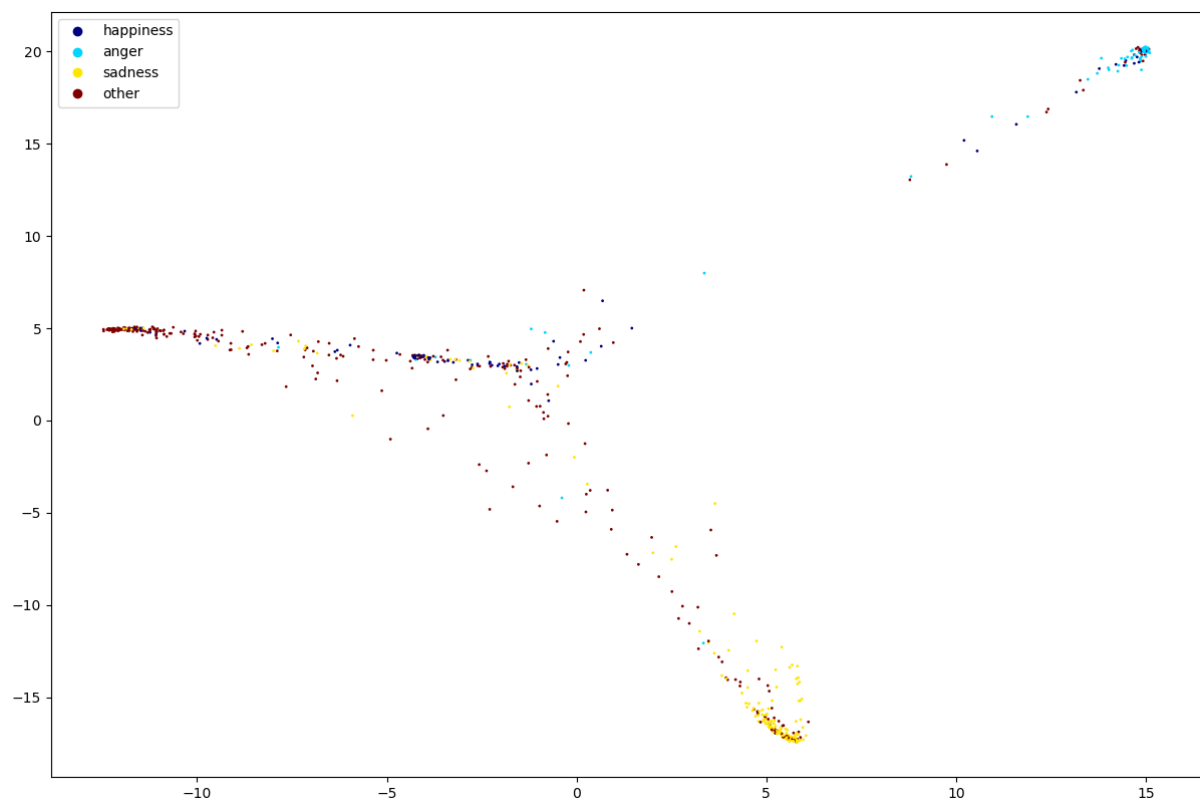
1.) Contrastive loss, Audeering model (frozen conv layers, lr = 1e-5, embedding_size = 128, batch_size = 12)

train accuracy = 0.946, test acc = 0.725, eval acc = 0.662

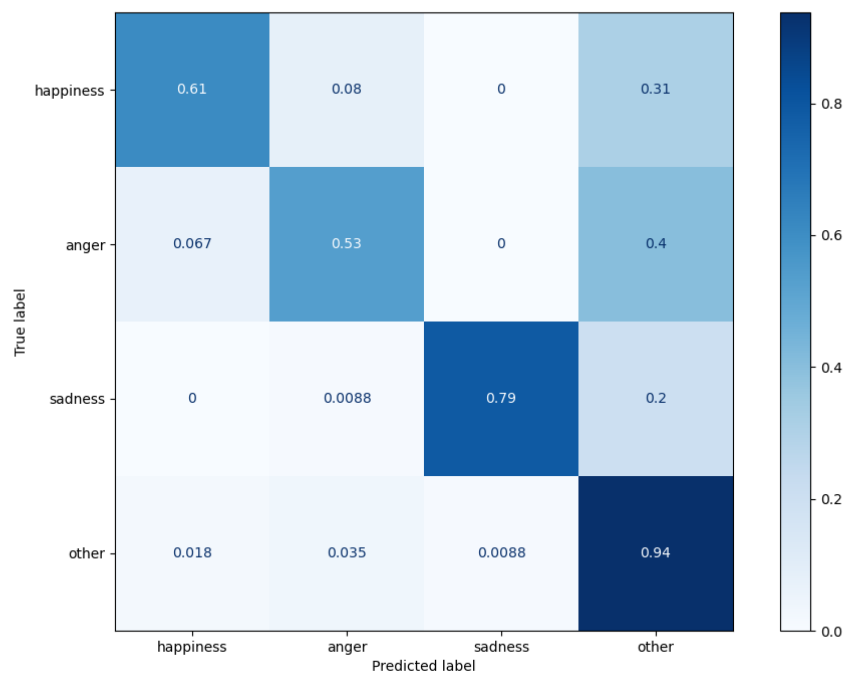
eval matrix:



eval embeddings:



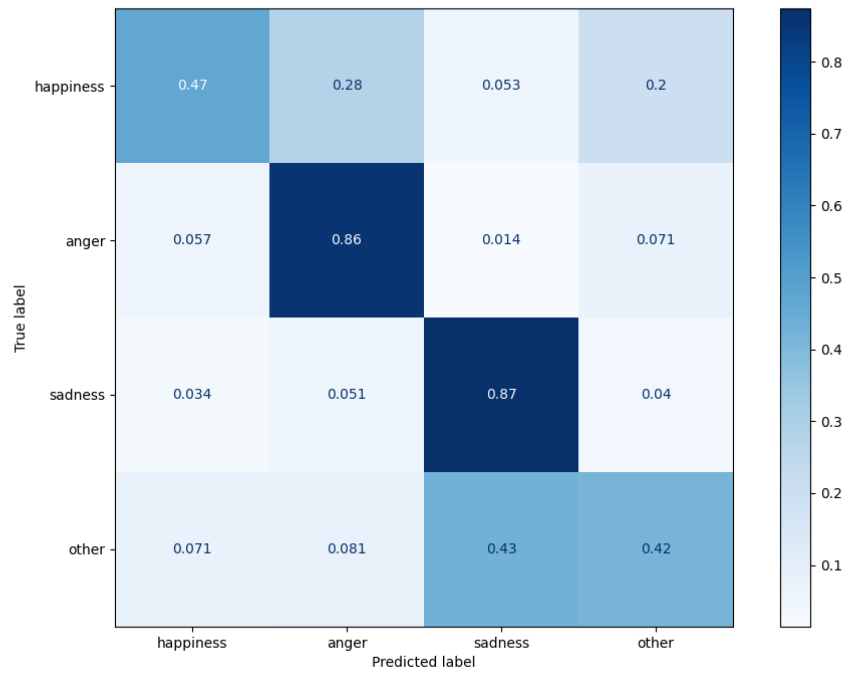
test matrix:



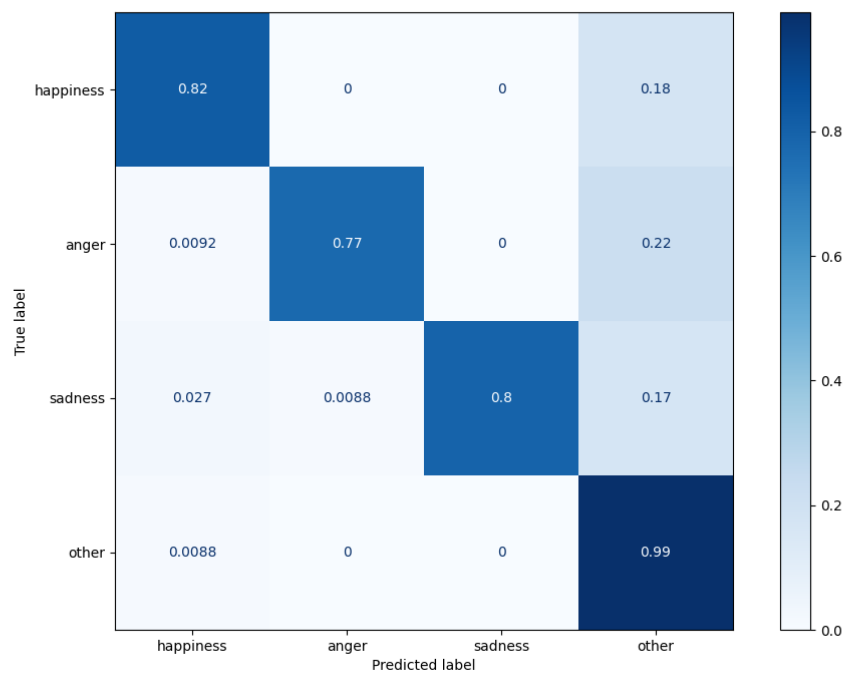
2.) Classification, Audeering model (frozen conv layers, lr = 1e-5, batch_size = 8)

train accuracy = 0.923, test acc = 0.882, eval acc = 0.597

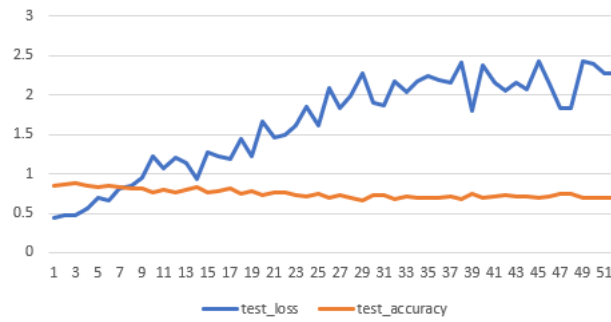
eval matrix:



test matrix:



loss plot:

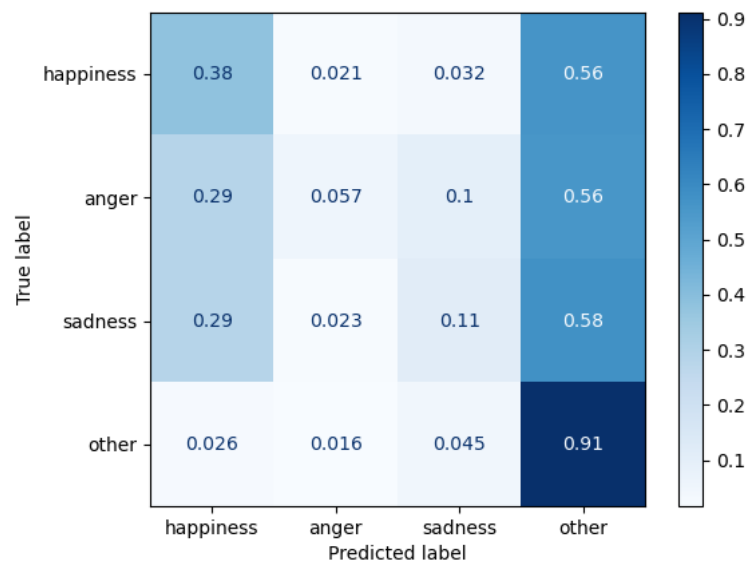


Alefiury model

<https://huggingface.co/alefiury/wav2vec2-xls-r-300m-pt-br-spontaneous-speech-emotion-recognition>

Predicts 3 classes: neutral, non-neutral male, non-neutral female

eval acc = 0.527



Laughter models

Newest models are very similar (483 - 494) (no extra other class, difference only a batch of laughter samples)

	A	B	C	D	E	F	G	H	I
1	number	confidence	total_accuracy	precision	f1	false pos	false neg	recall	
2	481	0.99	0.9	1	0.5	0	30	0.33	denoised
3	478	0.95	0.94	0.97	0.77	1	16	0.64	denoised
4	492	0.97	0.94	0.97	0.76	1	17	0.62	denoised
5	478	0.97	0.94	0.96	0.74	1	18	0.6	denoised
6	494	0.99	0.94	0.96	0.74	1	18	0.6	denoised
7	487	0.95	0.93	0.96	0.69	1	21	0.53	denoised
8	486	0.99	0.93	0.96	0.69	1	21	0.53	denoised
9	492	0.99	0.93	0.96	0.69	1	21	0.53	denoised
10	478	0.99	0.92	0.96	0.67	1	22	0.51	denoised
11	488	0.97	0.92	0.96	0.67	1	22	0.51	denoised
12	483	0.99	0.92	0.96	0.67	1	22	0.51	denoised
13	489	0.95	0.92	0.96	0.65	1	23	0.49	denoised
14	487	0.97	0.92	0.96	0.65	1	23	0.49	denoised
15	481	0.97	0.92	0.96	0.65	1	23	0.49	regular
16	479	0.97	0.92	0.95	0.63	1	24	0.47	denoised
17	485	0.99	0.92	0.95	0.63	1	24	0.47	denoised
18	489	0.97	0.91	0.95	0.61	1	25	0.44	denoised
19	488	0.99	0.91	0.95	0.61	1	25	0.44	denoised
20	493	0.99	0.91	0.95	0.56	1	27	0.4	denoised
21	479	0.99	0.9	0.94	0.54	1	28	0.38	denoised
22	481	0.99	0.9	0.94	0.54	1	28	0.38	regular
23	487	0.99	0.9	0.94	0.49	1	30	0.33	denoised
24	489	0.99	0.89	0.93	0.44	1	32	0.29	denoised
25	478	0.9	0.95	0.94	0.83	2	12	0.73	denoised
26	487	0.85	0.94	0.94	0.78	2	15	0.67	denoised
27	488	0.85	0.94	0.94	0.76	2	16	0.64	denoised
28	487	0.9	0.94	0.94	0.76	2	16	0.64	denoised
29	492	0.95	0.94	0.94	0.76	2	16	0.64	denoised
30	483	0.9	0.94	0.93	0.75	2	17	0.62	denoised
31	488	0.9	0.94	0.93	0.75	2	17	0.62	denoised
32	477	0.99	0.93	0.93	0.73	2	18	0.6	denoised
33	484	0.9	0.93	0.93	0.73	2	18	0.6	denoised
34	483	0.95	0.93	0.93	0.73	2	18	0.6	denoised
35	486	0.97	0.93	0.93	0.73	2	18	0.6	denoised
36	483	0.97	0.93	0.93	0.71	2	19	0.58	denoised
37	479	0.9	0.93	0.93	0.69	2	20	0.56	denoised
38	493	0.97	0.93	0.93	0.69	2	20	0.56	denoised
39	481	0.95	0.92	0.92	0.68	2	21	0.53	denoised
40	488	0.95	0.92	0.92	0.68	2	21	0.53	denoised
41	479	0.95	0.92	0.92	0.66	2	22	0.51	denoised
42	481	0.97	0.92	0.92	0.66	2	22	0.51	denoised
43	484	0.95	0.92	0.92	0.66	2	22	0.51	denoised
44	484	0.97	0.91	0.91	0.6	2	25	0.44	denoised
45	484	0.99	0.9	0.89	0.53	2	28	0.38	denoised
46	487	0.75	0.96	0.92	0.84	3	10	0.78	denoised
47	479	0.75	0.95	0.91	0.8	3	13	0.71	denoised
48	479	0.8	0.95	0.91	0.8	3	13	0.71	denoised

with added other (relabelled false positives)

1339	496	0.55	0.87	0.53	0.66	36	5	0.89	regular
1340	496	0.55	0.92	0.71	0.76	15	8	0.82	denoised
1341	496	0.6	0.89	0.58	0.7	29	5	0.89	regular
1342	496	0.6	0.93	0.76	0.79	12	8	0.82	denoised
1343	496	0.65	0.9	0.62	0.72	23	7	0.84	regular
1344	496	0.65	0.94	0.82	0.81	8	9	0.8	denoised
1345	496	0.7	0.91	0.69	0.73	16	10	0.78	regular
1346	496	0.7	0.95	0.87	0.81	5	11	0.76	denoised
1347	496	0.75	0.93	0.77	0.76	10	11	0.76	regular
1348	496	0.75	0.95	0.94	0.79	2	14	0.69	denoised
1349	496	0.8	0.93	0.79	0.77	9	11	0.76	regular
1350	496	0.8	0.94	0.97	0.77	1	16	0.64	denoised
1351	496	0.85	0.94	0.81	0.78	8	11	0.76	regular
1352	496	0.85	0.93	0.96	0.72	1	19	0.58	denoised
1353	496	0.9	0.93	0.8	0.75	8	13	0.71	regular
1354	496	0.9	0.92	0.96	0.65	1	23	0.49	denoised
1355	496	0.95	0.94	0.89	0.79	4	13	0.71	regular
1356	496	0.95	0.92	0.95	0.63	1	24	0.47	denoised
1357	496	0.97	0.94	0.94	0.76	2	16	0.64	regular
1358	496	0.97	0.91	0.95	0.56	1	27	0.4	denoised
1359	496	0.99	0.92	0.92	0.68	2	21	0.53	regular
1360	496	0.99	0.89	0.93	0.44	1	32	0.29	denoised