Nkululeko

a form based speech machine learning tool

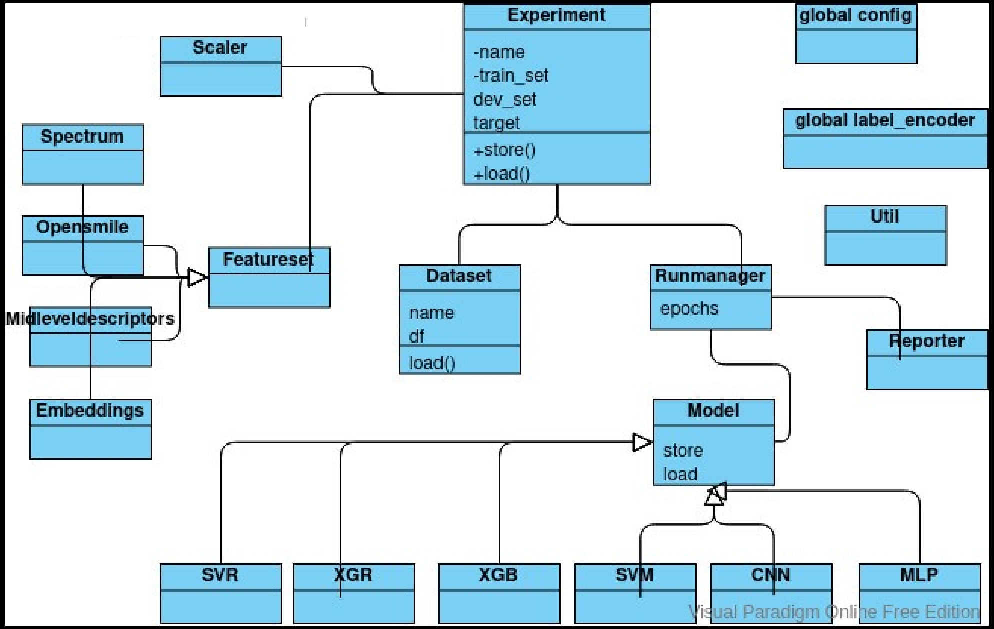
Felix Burkhardt

outline

- what is Nkululeko
- how to use it
- example experiments

what is Nkululeko?

- a software written in Python hosted on github*
- a tool to do machine learning (ML) experiments WITHOUT the need to program yourself
- focused on combinations of features and machine learners
- uses configuration file templates described in a blog**



* https://github.com/felixbur/nkululeko

** blog.syntheticspeech.de/?s=nkululeko

motivation

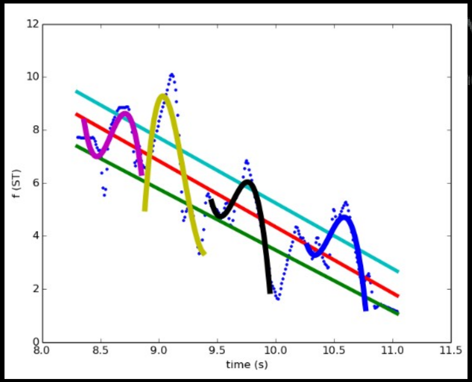
- with the success of Deep Learning, machine learning dominates science
- empiricists sometimes struggle with programming
- teaching students
- re-use of code

features

Three kinds:

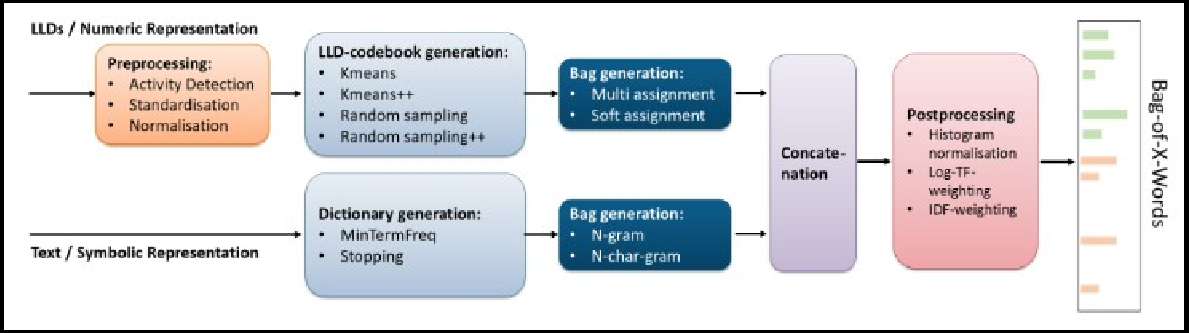
- expert features
- brute force (needs feature selection)
- learned
  - autoencoders
  - embeddings, latent space

features



img src: [2]

- opensmile [1]
  - (e)GeMAPS – 62/88
  - Compare16 - 6,373
- mid level descriptors [2]
- openXbow [3]



LLDs / Numeric Representation

Preprocessing:
- Activity Detection
- Standardisation
- Normalisation

LLD-codebook generation:
- Kmeans
- Kmeans++
- Random sampling
- Random sampling++

Bag generation:
- Multi assignment
- Soft assignment

Text / Symbolic Representation

Dictionary generation:
- MinTermFreq
- Stopping

Bag generation:
- N-gram
- N-char-gram

Concate-nation

Postprocessing
- Histogram normalisation
- Log-TF-weighting
- IDF-weighting

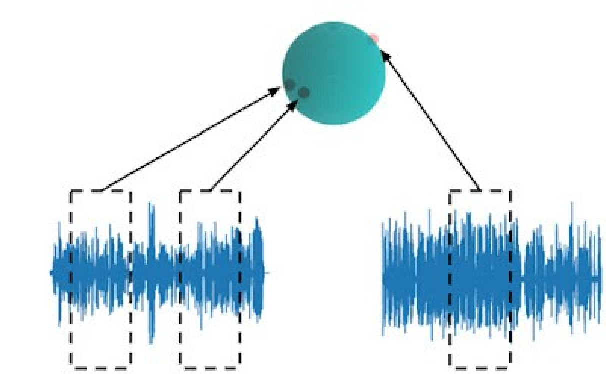
Bag-of-X-Words

img src: [3]

[1] Eyben, F., Wöllmer, M. and Schuller, B. (2010). opensmile – the munich versatile and fast open-source audio feature extractor.
[2] Reichel, U., Triantafyllopoulos, A., Oates, C., Huber,S., and Schuller, B. (2020). Spoken language iden-tification by means of acoustic mid-level descriptors
[3] Schmitt, M. and Schuller, B. (2017). openxbow - introducing the passau open-source crossmodal bag-of-words toolkit.

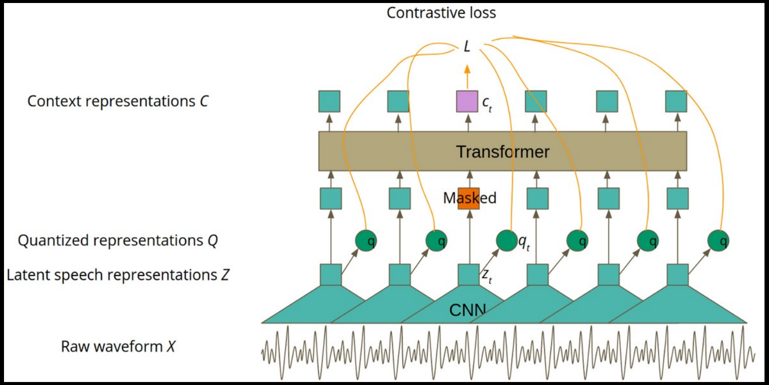
features cont.

- Logmel spectrograms
- TRILL [1]
- Wav2vec 2.0 [2]



TRILL

img src: https://ai.googleblog.com/2020/06/improving-speech-representations-and.html


Contrastive loss
L
Context representations C
c_t
Transformer
Masked
Quantized representations Q
q
q_t
Latent speech representations Z
z_t
CNN
Raw waveform X

Wav2Vec 2.0

img src: https://towardsdatascience.com/wav2vec-2-0-a-framework-for-self-supervised-learning-of-speech-representations-7d3728688cae

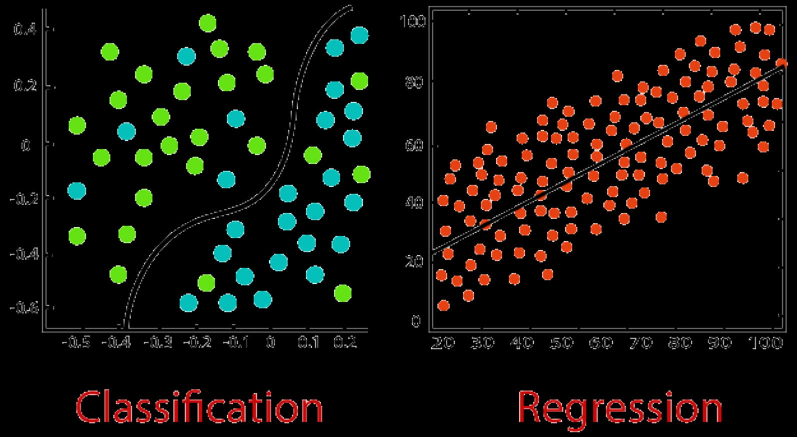
[1] Shor, J., Jansen, A., Maor, R., Lang, O., Quitry, F.,Tagliasacchi, M., Tuval, O., Shavitt, I., Emanuel, D.,and Haviv, Y. (2020). Towards learning a universalnon-semantic representation of speech

[2] Baevski, A., Zhou, Y., Mohamed, A., and Auli,M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representation

machine learners

Two distinctions

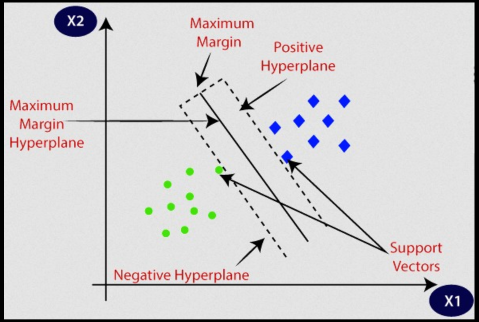
- classifiers versus regressors
  - classifier: prob. of a specific category
  - regressor: predict a scalar
- approach
  - geometric
  - decision trees
  - ANNs
  - ...



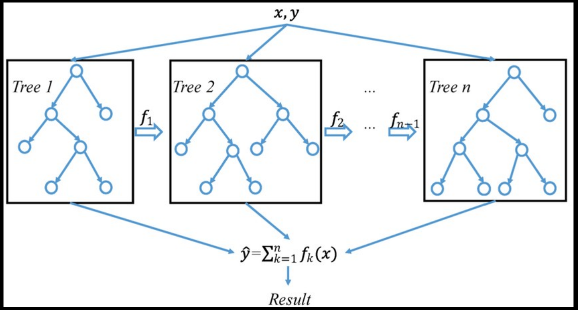
Img. Src: https://www.javatpoint.com/regression-vs-classification-in-machine-learning

learners

- SVM: support vector machine
- SVR
- XGB: XG-boost
- XGR



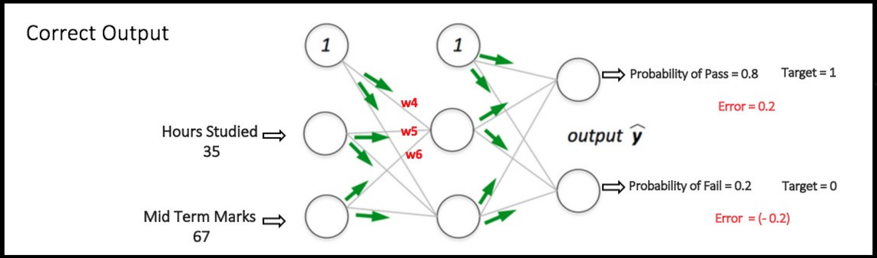
img src: https://medium.com/@skilltohire/support-vector-machines-4d28a427ebd



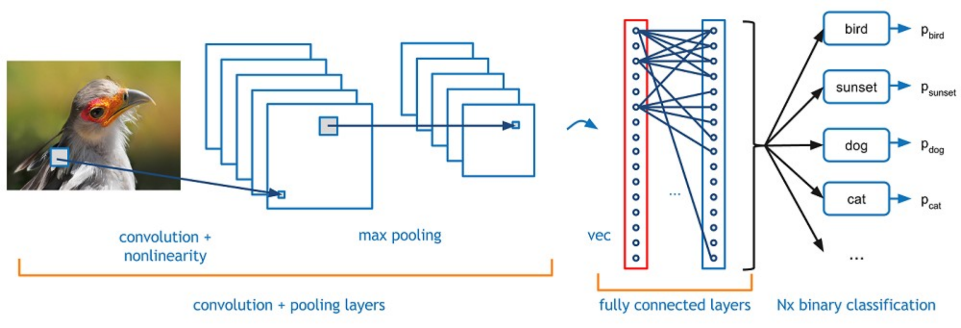
img src: Wang, Yuanchao & Pan, Z. & Zheng, J. & Qian, L. & Mingtao, Li. (2019).
A hybrid ensemble method for pulsar candidate classification.

learners cont.

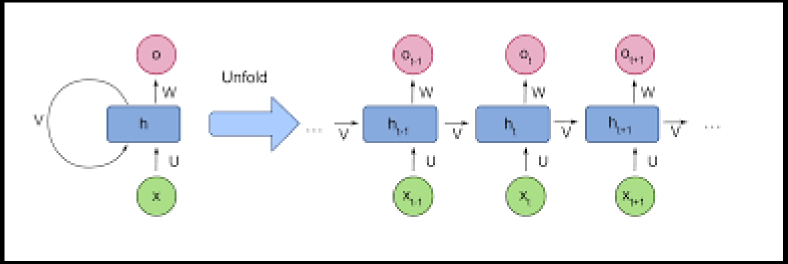
- MLP: multi layer perceptron
- CNN: convolutional neural net
- RNN: recurrent neural net
- Transformers



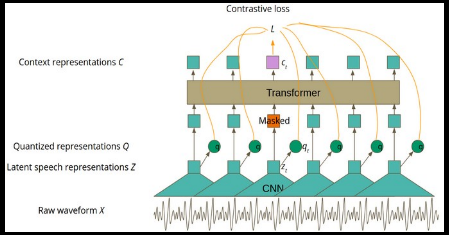
img src: https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/



img src: https://medium.datadriveninvestor.com/convolution-neural-network-22565e6d8156



img src: Wikimedia


Contrastive loss
L
Context representations C
c_t
Transformer
Masked
Quantized representations Q
q_t
Latent speech representations Z
z_t
CNN
Raw waveform X

the configuration file

- Key-value pairs
- Organized in sections
  - EXP
  - DATA
  - FEAT
  - MODEL
  - PLOT

```
[EXP]
root = ./tests/
name = exp_syntact
runs = 1
epochs = 1
save = True
[DATA]
databases = ['syntact']
syntact = /home/felix/data/research/syntAct/syntact/
syntact.split_strategy = speaker_split
syntact.testsplit = 50
syntact.value_counts = True
target = emotion
labels = ['angry', 'happy', 'neutral', 'sad']
[FEATS]
#type = trill
type = os
scale = standard
[MODEL]
type = svm
save = True
[PLOT]
value_counts = True
tsne = True
```

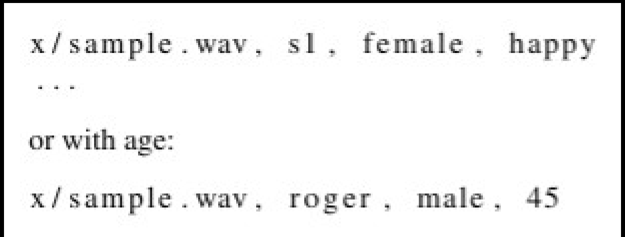
conf. file EXP section

- name
- type [classification | regression]
- #epochs
- #runs

https://github.com/felixbur/nkululeko/blob/main/ini_file.md

conf. file DATA section

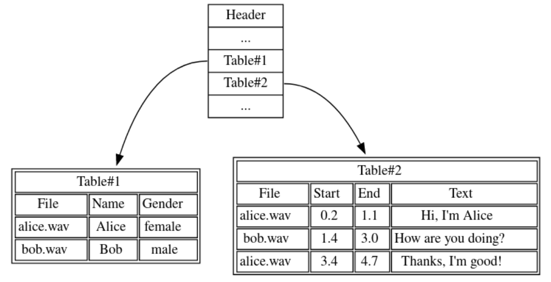
- databases
  - type: [audformat* | CSV]
  - table specifics
  - train/test splits
- type [cross corpus | train-test]
- trains, tests
- label mapping
- binning (scalar → classes)
- sex: data filter



```
x/sample.wav, s1, female, happy
...
```

or with age:

```
x/sample.wav, roger, male, 45
```



Header
...
Table#1
Table#2
...

Table#1

File	Name	Gender
alice.wav	Alice	female
bob.wav	Bob	male

Table#2

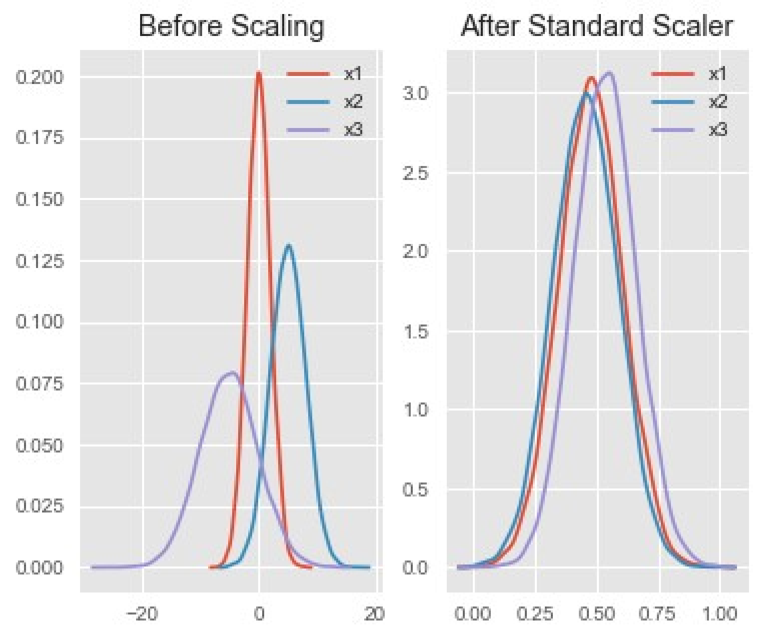
File	Start	End	Text
alice.wav	0.2	1.1	Hi, I'm Alice
bob.wav	1.4	3.0	How are you doing?
alice.wav	3.4	4.7	Thanks, I'm good!

*https://audeering.github.io/audformat/

https://github.com/felixbur/nkululeko/blob/main/ini_file.md

conf. file FEATS section

- type: [os | spectra | mld | xbow | trill | wav2vec]
- scale: [std | spkr | sex]
- model: path to model



Img src: https://shauryauppal.medium.com/how-and-where-to-apply-feature-scaling-machine-learning-93316663cd63

https://github.com/felixbur/nkululeko/blob/main/ini_file.md

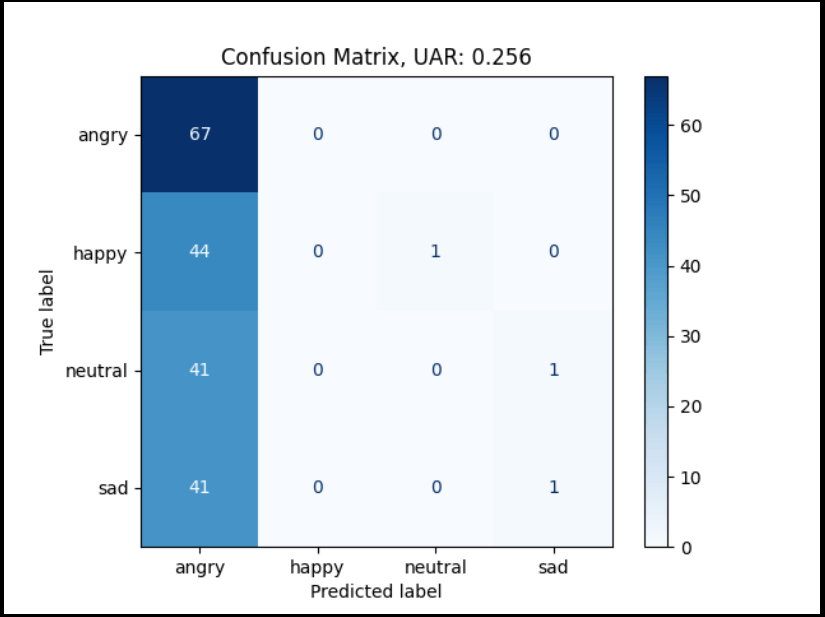
conf. file MODEL section

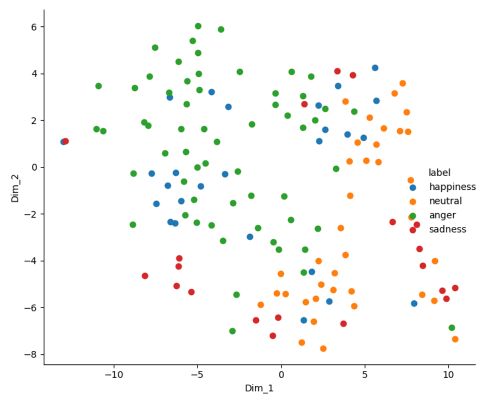
- type: [svm | svr | xgb | xgr | mlp | mlp-reg | cnn]
- tuning_params: 5 fold cross optimization
- layers, loss-function, learning rate: ANN specs
- class_weight

https://github.com/felixbur/nkululeko/blob/main/ini_file.md

conf. file PLOT   section

- plot_epochs
- plot_anim_progression
- plot_epoch_progression
- plot_best_model
- t-SNE

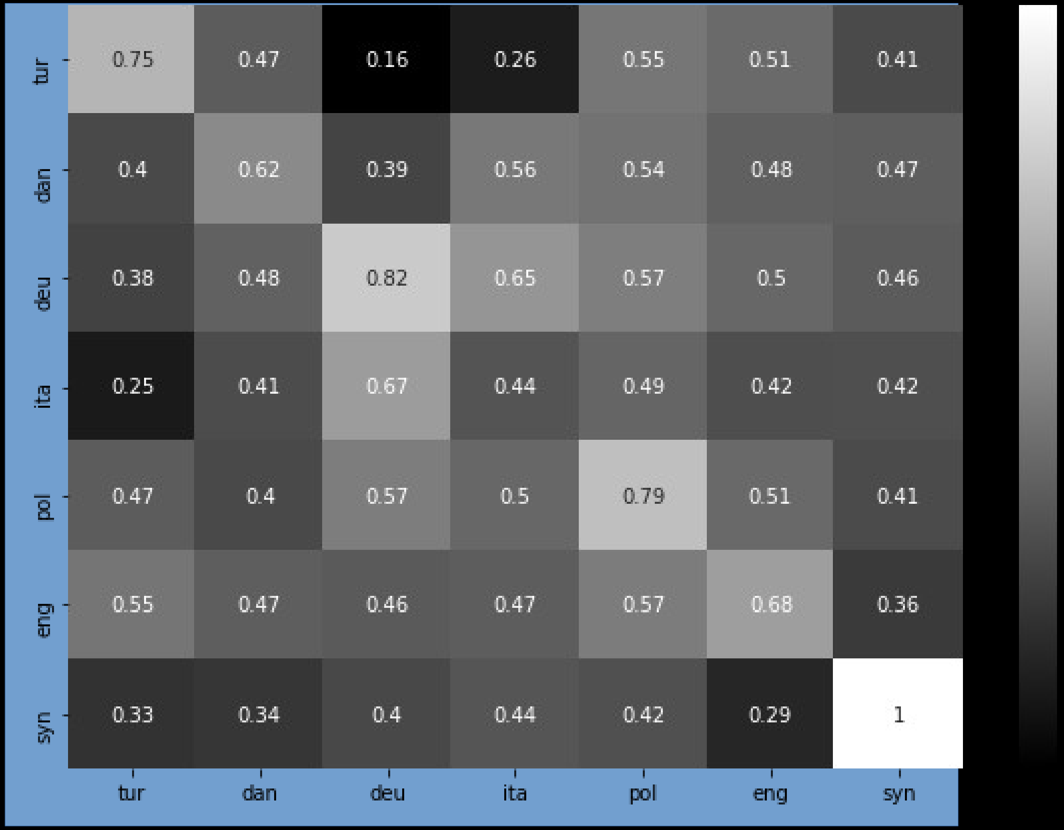




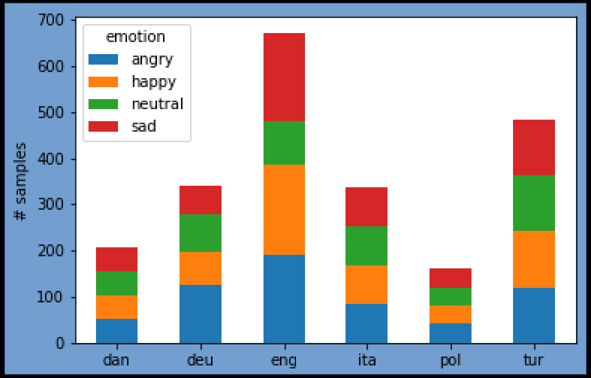
https://github.com/felixbur/nkululeko/blob/main/ini_file.md

experiments

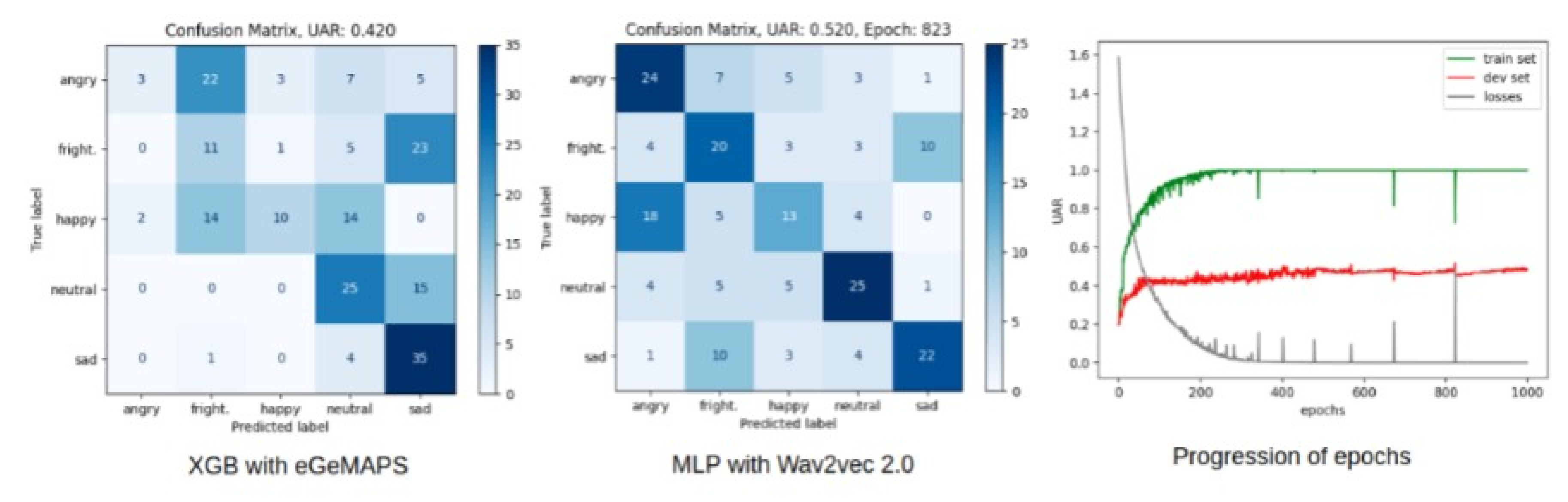
- 6 European acted emotional databases + synthesized emotion



	tur	dan	deu	ita	pol	eng	syn
tur	0.75	0.47	0.16	0.26	0.55	0.51	0.41
dan	0.4	0.62	0.39	0.56	0.54	0.48	0.47
deu	0.38	0.48	0.82	0.65	0.57	0.5	0.46
ita	0.25	0.41	0.67	0.44	0.49	0.42	0.42
pol	0.47	0.4	0.57	0.5	0.79	0.51	0.41
eng	0.55	0.47	0.46	0.47	0.57	0.68	0.36
syn	0.33	0.34	0.4	0.44	0.42	0.29	1



experiments cont.



Confusion Matrix, UAR: 0.420

True label \ Predicted label	angry	fright.	happy	neutral	sad
angry	3	22	3	7	5
fright.	0	11	1	5	23
happy	2	14	10	14	0
neutral	0	0	0	25	15
sad	0	1	0	4	35

XGB with eGeMAPS

Confusion Matrix, UAR: 0.520, Epoch: 823

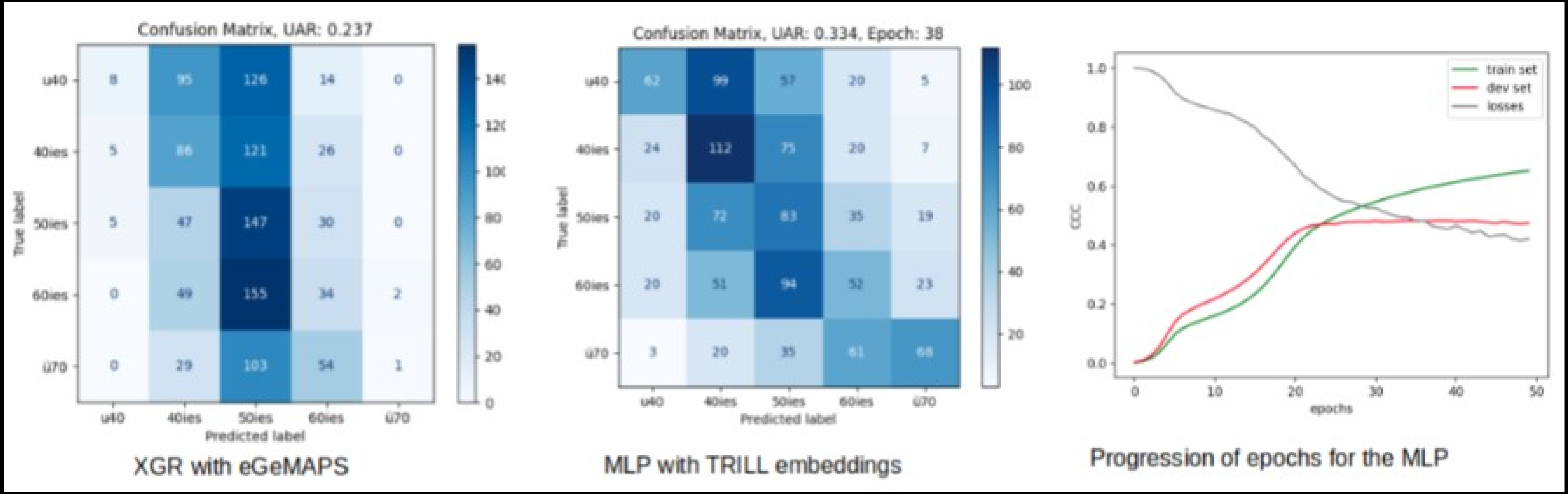
True label \ Predicted label	angry	fright.	happy	neutral	sad
angry	24	7	5	3	1
fright.	4	20	3	3	10
happy	18	5	13	4	0
neutral	4	5	5	25	1
sad	1	10	3	4	22

MLP with Wav2vec 2.0

Progression of epochs

- comparing expert with learned features on cross databases acted emotion learning
- Berlin EmoDB vs. Polish data

experiments cont.



Confusion Matrix, UAR: 0.237

True label \ Predicted label	u40	40ies	50ies	60ies	ü70
u40	8	95	126	14	0
40ies	5	86	121	26	0
50ies	5	47	147	30	0
60ies	0	49	155	34	2
ü70	0	29	103	54	1

XGR with eGeMAPS

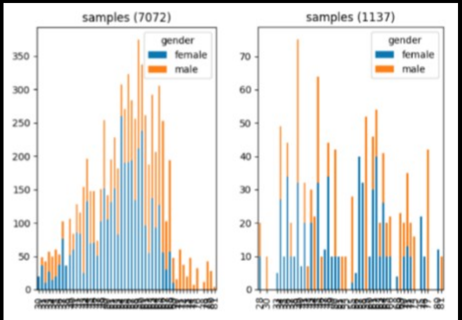
Confusion Matrix, UAR: 0.334, Epoch: 38

True label \ Predicted label	u40	40ies	50ies	60ies	ü70
u40	62	99	57	20	5
40ies	24	112	75	20	7
50ies	20	72	83	35	19
60ies	20	51	94	52	23
ü70	3	20	35	61	68

MLP with TRILL embeddings

Progression of epochs for the MLP

- comparing expert with learned features on age regression
- Data: German parliament data



train and test distribution

wrap up

- introduced Nkululeko
- a software to do machine learning experiments on spoken data without programming
- combines features and learners
- available open source with MIT license