DTU Compute
Mads Esben Hansen, s174434
Ida Bukh Villesen, s174385

Anomaly Detection in Time Series data using Autoencoders

Deep Learning 02456, E21

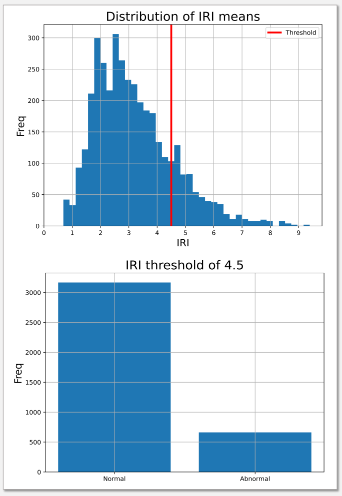
Aim of project and Data Processing

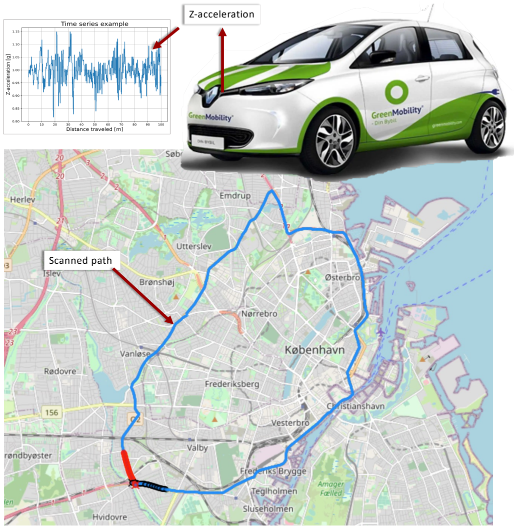
Aim: Identify rough roads from Z-acceleration monitors in Green Mobility cars
- Z-acceleration signal as Time Series, with varying length as a result of changing speed, covering 100 m each
- Roughness measured by the International Roughness Index (IRI)
- IRI > 4.5 → Bad road

Method: Autoencoder using RNNs to reconstruct signals from *good* roads → fail at *bad* roads
- Gated Recurrent Unit (GRU) and Dense Layers

Data pre-processing:
- 0-padding of Time Series to adjust and equalize lengths, using only the 80% shortest series
- Longest 20% of data cut to fit 0-padded data
- Final lengths of series → 3366 indices each

Splitting the data into sets:
- ~ 80% good road → split further into 80/20 train/test set
- ~ 20% bad road → test data, tested on final network

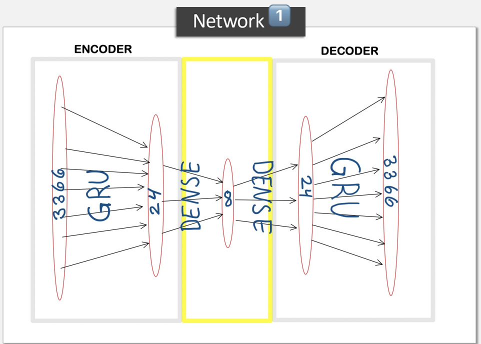




Models and Networks

Autoencoder: Using latent space representation
- Trained to reconstruct good roads
- Good reconstruction of good roads (IRI < 4.5)
- Fails reconstructing bad roads (unseen signal pattern)

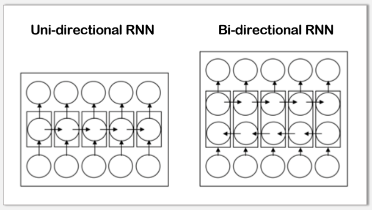
Custom Loss Function for optimization:
- MSE loss function
- MAE → for robust estimations
- Ignoring 0-padding for optimal training of network



Network 1:
- Unsupervised Autoencoder
- Uni-directional *GRU*
- *Dense layer* in AE bottleneck

Network 2:
- Unsupervised Autoencoder
- Bi-directional *GRU*
- *Dense layer* to output layer

Network 3:
- Supervised network
- Input layer Uni-directional GRU
- 2x Dense Layers connection to output dimension 1

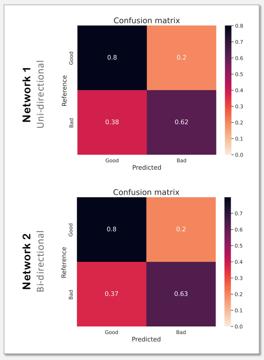


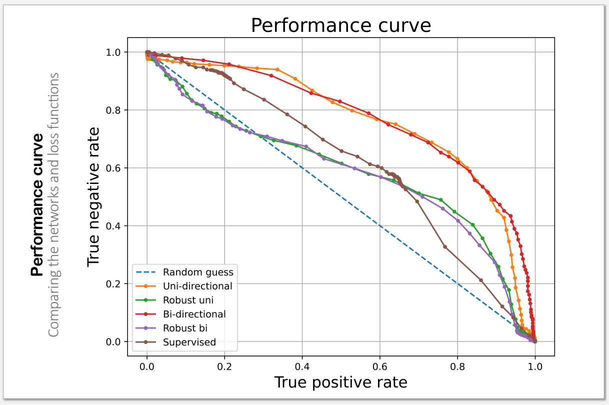
Results

Performance curve:
- Comparison of networks
- Threshold neutral

Confusion matrices:
- Performance of networks - thresholded at 80% true positive

Optimized parameters	Network 1	Network 2	Network 3
Batch size	32	32	256
Optimizer	RMSprop	RMSprop	RMSprop
Dropout	0.1	No	No
Learning rate	1e-3	1e-3	1e-3
Weight decay	1e-4	1e-4	1e-4





Conclusion and remarks

The aim: Identify rough roads from Z-acceleration monitors in Green Mobility cars
- Implemented fully functioning AE with GRU layers using Z-acceleration
- Successfully trained to reconstruct goad road
- 80% correct identification of good road & 63% of bad road at optimum
- Multiple architectures implemented and tested for optimal model performance

Remarks: Topics for discussion
- All networks sensitive to learning rates → gets stuck non-optimal local minima
- Sub-optimal performance of supervised and robust networks (unidentified optimization problems?)
- Possible to increase performance by including speed as an input additional to the z-acceleration?
- Alternative method to data pre-processing with 0-padding → network flexible to handle varying signal input sizes?

References

- Road Roughness Estimation Using Machine Learning, https://arxiv.org/abs/2107.01199
- Time Series Reconstruction using a Bidirectional Recurrent Neural Network based Encoder-Decoder Scheme. https://www.researchgate.net/publication/333145226_Time_Series_Reconstruction_Using_a_Bidirectional_Recurrent_Neural_Network_based_Encoder-Decoder_Scheme
- https://towardsdatascience.com/building-rnn-lstm-and-gru-for-time-series-using-pytorch-a46e5b094e7b
- https://discuss.pytorch.org/t/gru-time-series-autoencoder/77126
- https://towardsdatascience.com/understanding-bidirectional-rnn-in-pytorch-5bd25a5dd66
- https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21