

BerDiBa



Phase 2

BERLINER DIGITALER BAHNBETRIEB

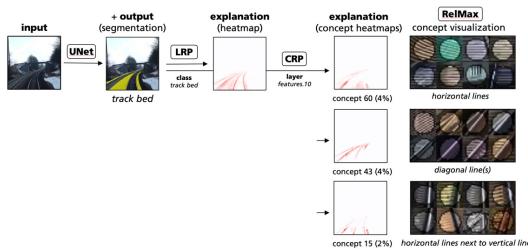


Gliederung

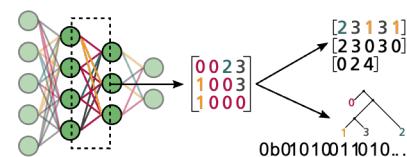
- Firmenvorstellung Fraunhofer HHI und Beiträge zum BerDiBa Projekt
- Meilensteinpräsentation
 - XAI-getriebene NN Kompression
 - Rationale hinter Erklärbarkeit und Kompression
 - XAI-korrigierte Quantisierung und Filter-Sparsifikation
 - Mixture-of-Relevant-Experts
 - Präsentation des Demonstrators
- Zusammenfassung

Themen / Projektinhalte

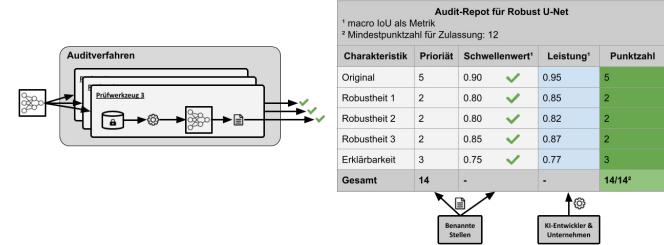
Erklärbarkeit und Interpretierbarkeit von KI (XAI)



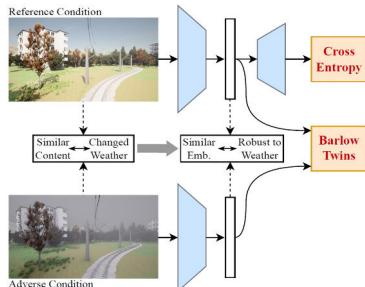
(XAI-) Kompression und Effizienzsteigerung neuronaler Netze



KI Zertifizierbarkeit, Prüfwerkzeuge, Audit-Verfahren



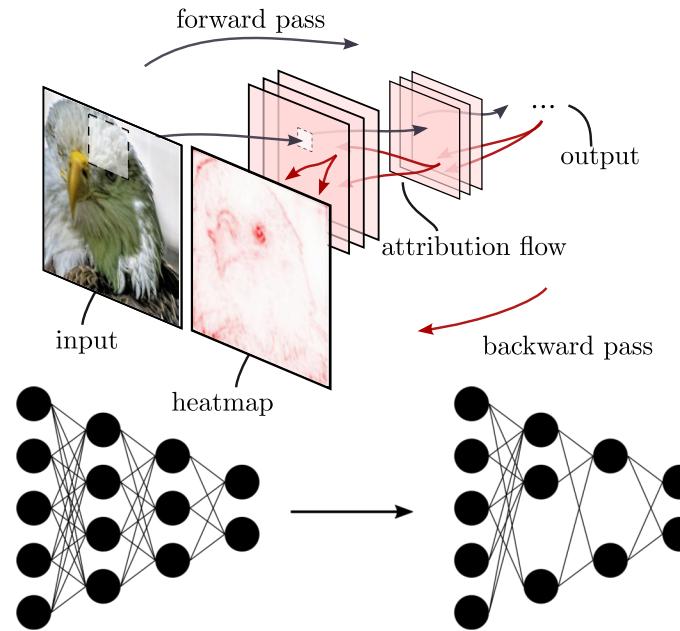
Robustheit von Computer Vision Verfahren



Steigerung der Detektionsgüte durch Datensynthese und multimodales Training

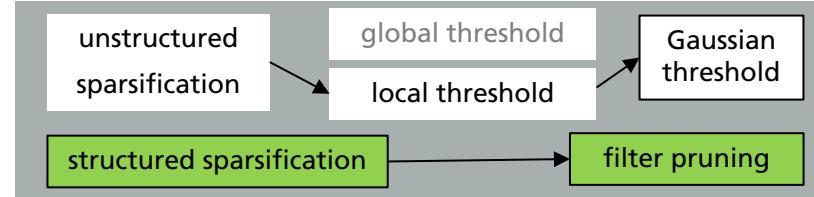
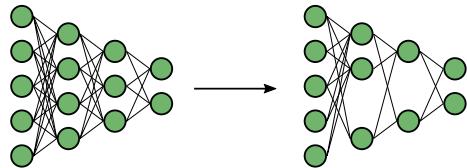


Evaluierung der XAI Kompression

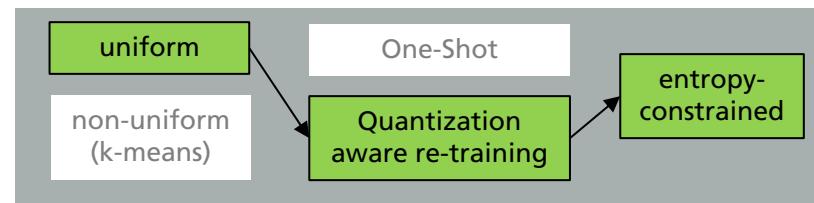
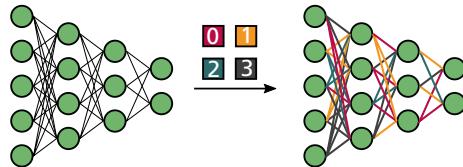


Kompressions-Pipeline / Identifikation bester Kandidaten

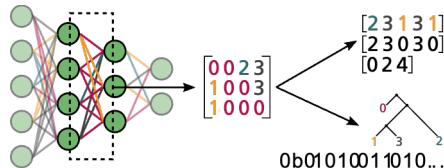
I. Parameter-/ Operationsreduktion:



II. Präzisionsreduktion der Operanden:



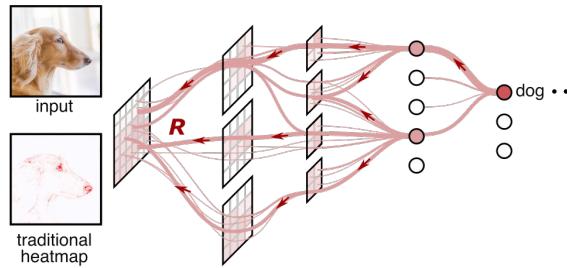
III. Kodierung (verlustfrei):



XAI in der semantischen Segmentierung

- Layer-wise Relevance Propagation (LRP^[7]) / Concept Relevance Propagation (CRP^[8])

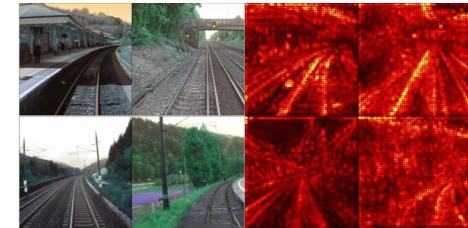
a traditional explanation (LRP)



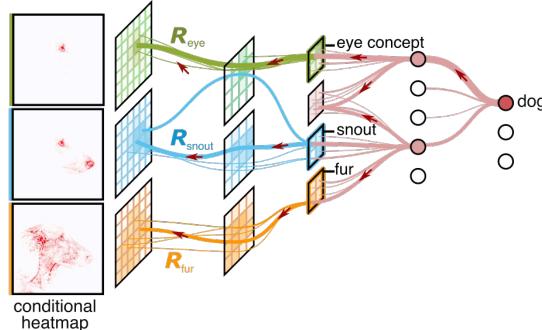
Semantische Segmentierung

region of interest:
prädierte
Segmentierungs-
maske

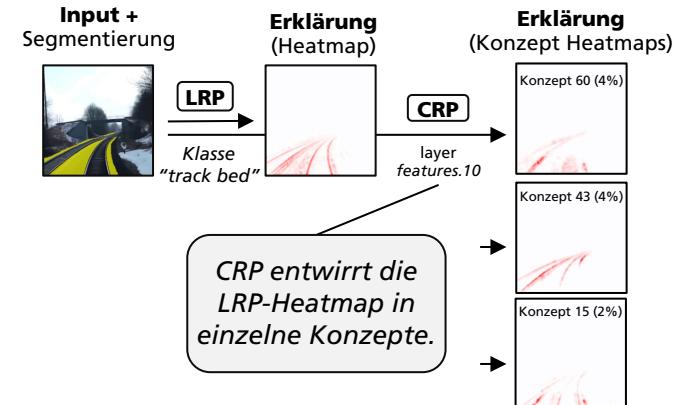
LRP Beispiele:



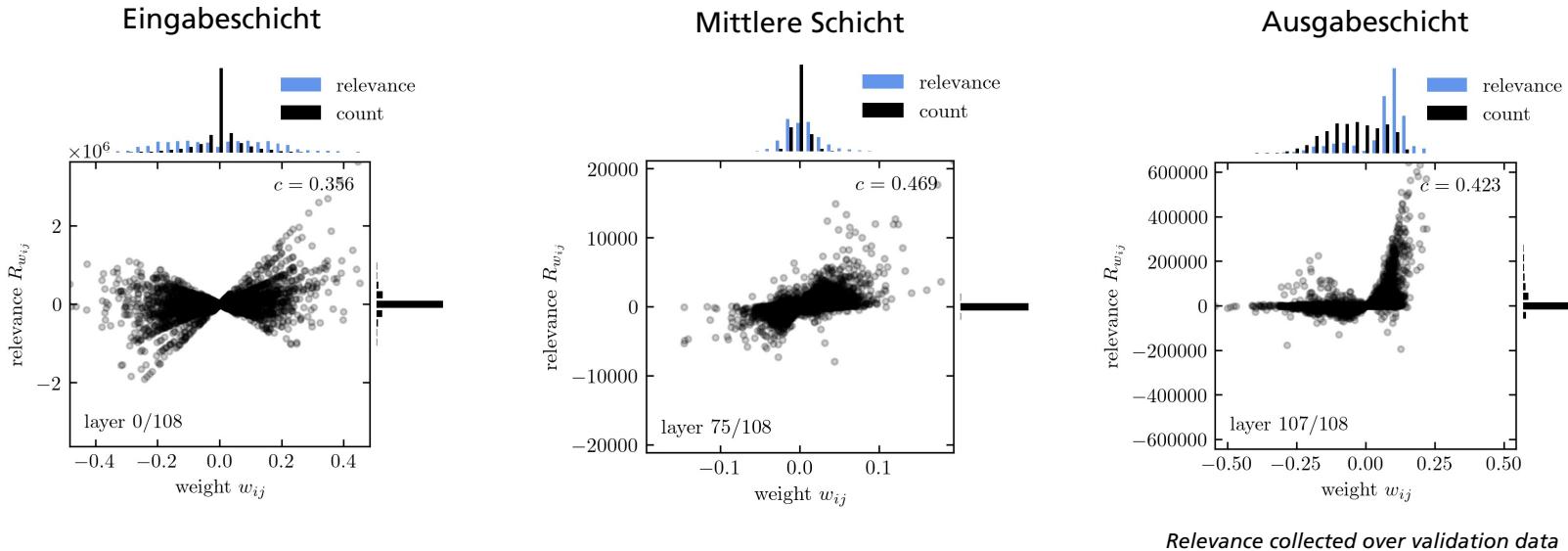
b concept-conditioned explanation (CRP)



CRP Beispiele:



Rationale hinter Erklärbarkeit und Kompression



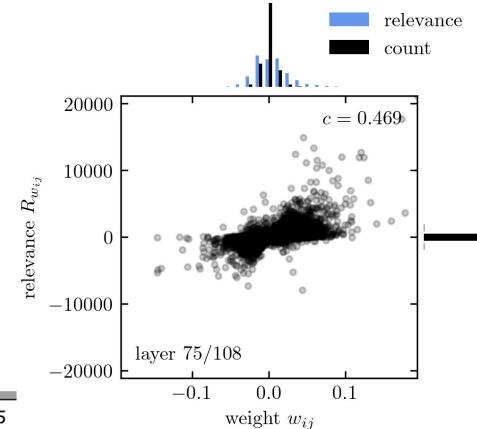
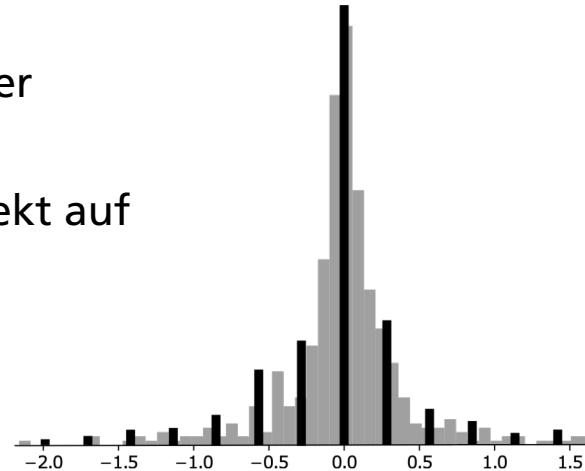
- (1) Gewichte mit hohen Werten können „irrelevant“ sein
- (2) Kleine Gewichtswerte können sich als relevant erweisen

XAI-corrected Entropy-Constrained Quantization (ECQ^X [6])

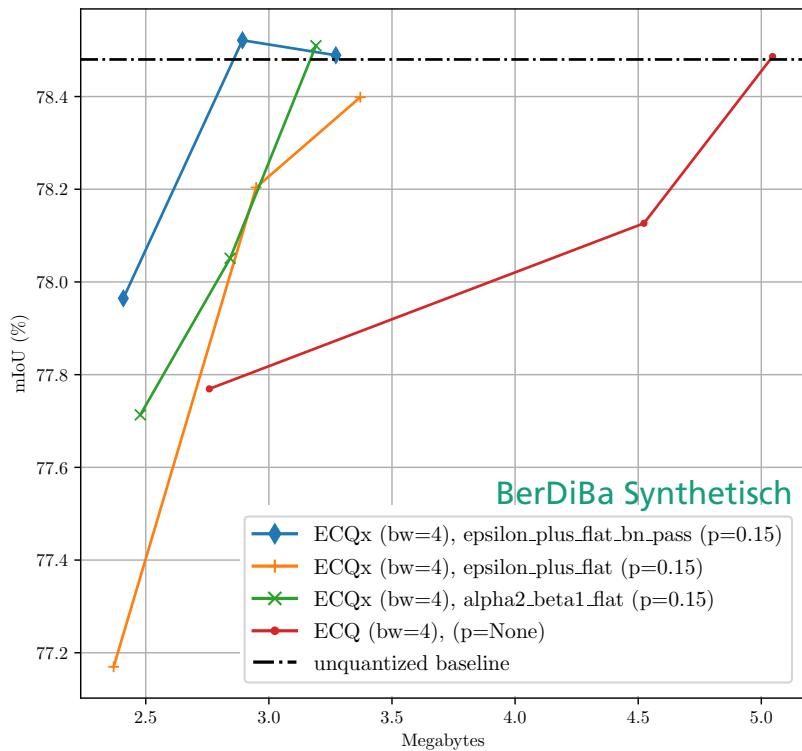
- Neue Zuweisungsfunktion:

$$A^{(l)}(\mathbf{w}^{(l)}) = \underset{c}{\operatorname{argmin}} \begin{cases} \rho R^{(l)} \cdot [d(\mathbf{w}^{(l)}, \mathbf{w}_0^{(l)}) - \lambda^{(l)} \log_2(P_0^{(l)})], & \text{if } c = 0 \\ d(\mathbf{w}^{(l)}, \mathbf{w}_c^{(l)}) - \lambda^{(l)} \log_2(P_c^{(l)}), & \text{if } c \neq 0 \end{cases}$$

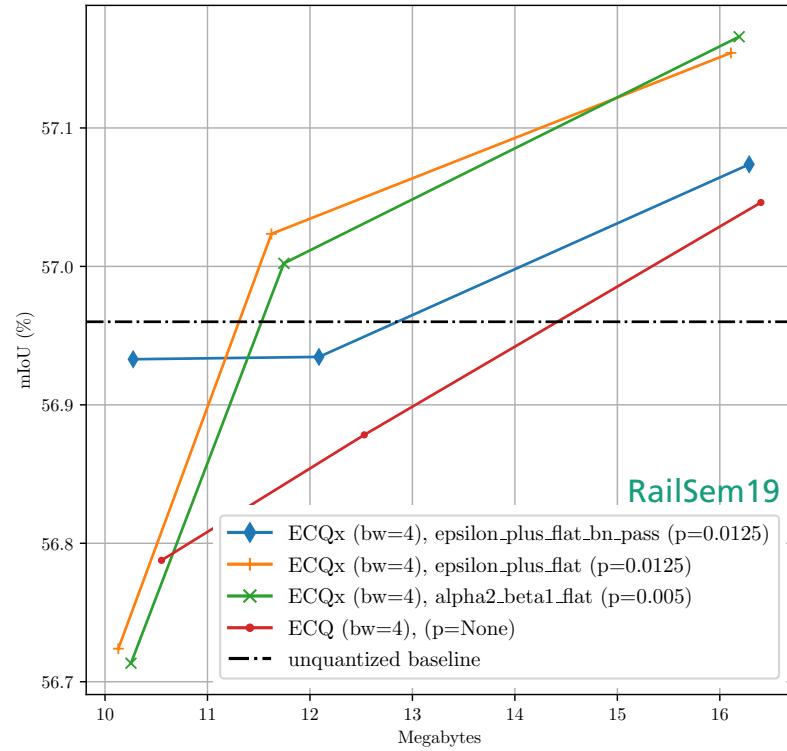
- Relevanz-Korrektur der Zuweisungsfunktion
- Regularisierender Effekt auf das Training der NN-Quantisierung



ECQ^X Resultate – Speichergröße der enkodierten Bitströme



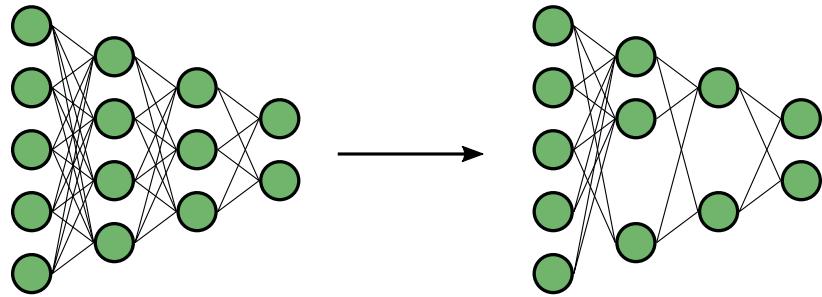
BerDiBa Synthetisch



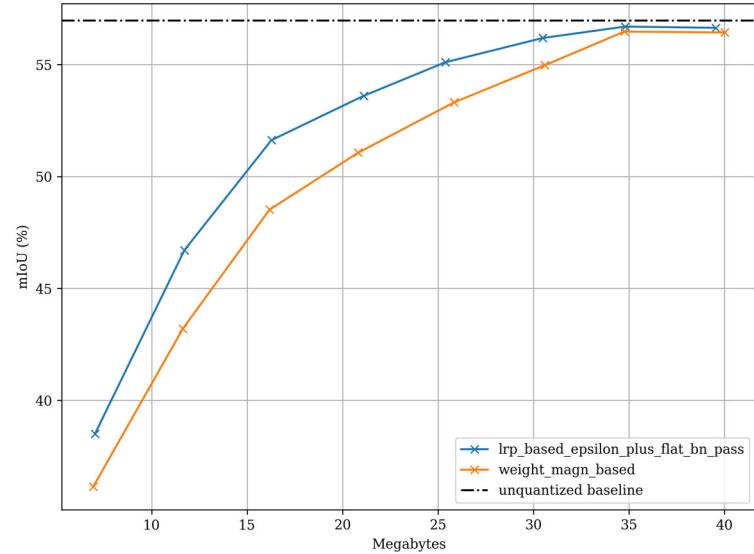
RailSem19

original size: 235MB

Pruning / Strukturierte Sparsifikation

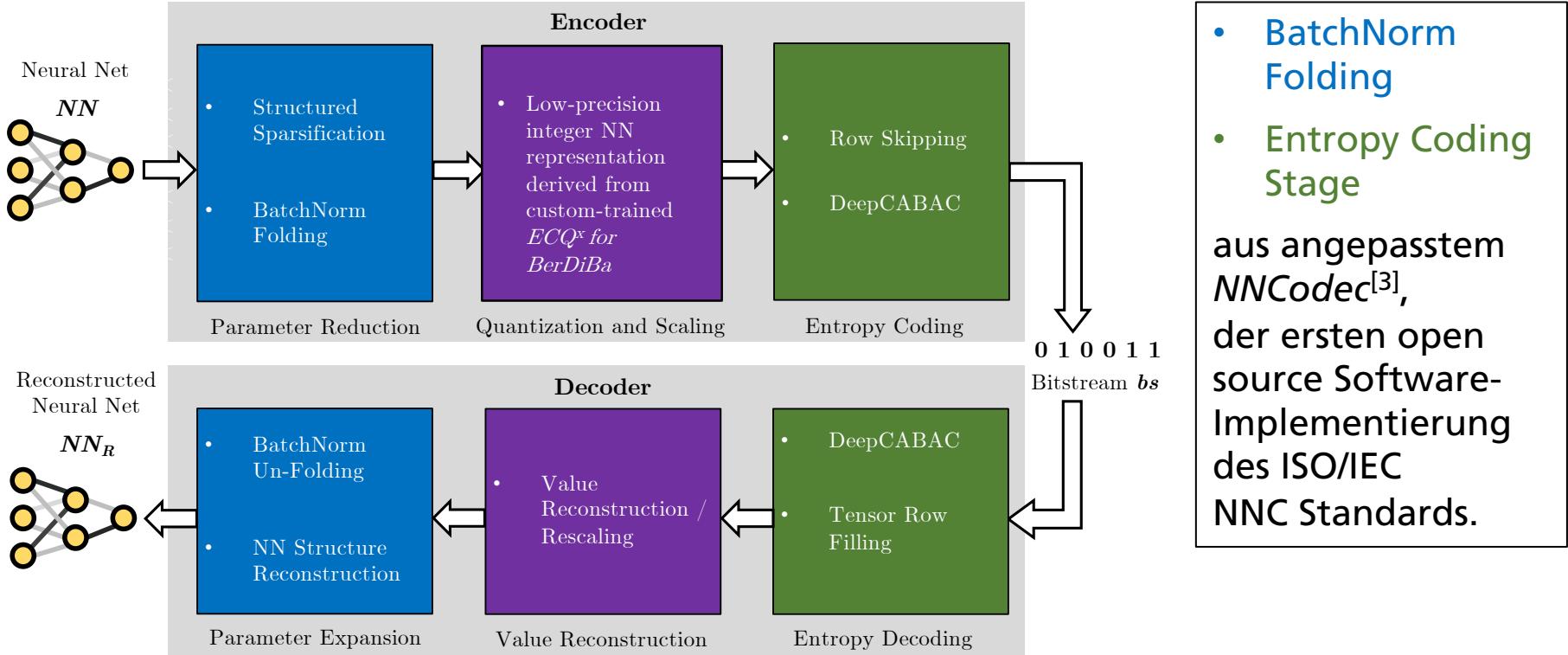


- Basierend auf einem Kriterium / Grenzwert werden
 - ganze Neuronen (samt aller verbundener Eingabegewichte) bei linearen Schichten
 - ganze Ausgabe-Channel (sog. „Filter“) bei Konvolutionsschichten
- gelöscht / zu Null gesetzt.



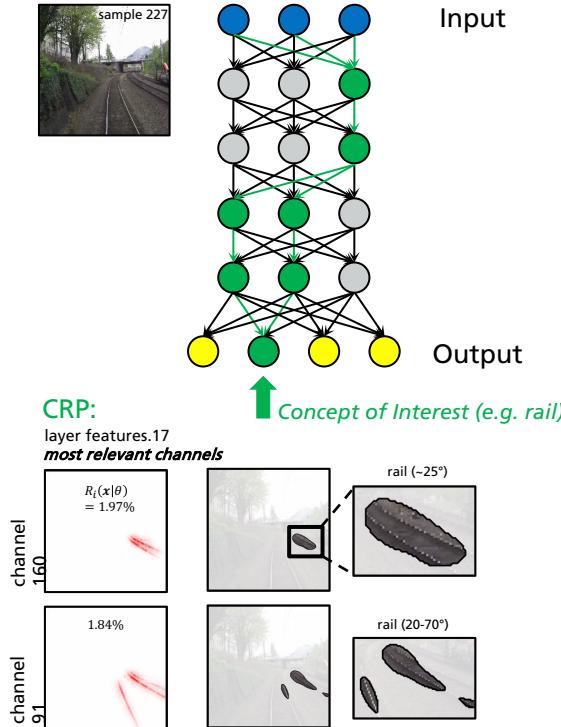
➤ RailSem19 Beispiel: die **LRP Relevanz** als **Pruning-Kriterium** erzielt bessere Performanz als der **Gewichtswert**

Encoder-Decoder Aufbau

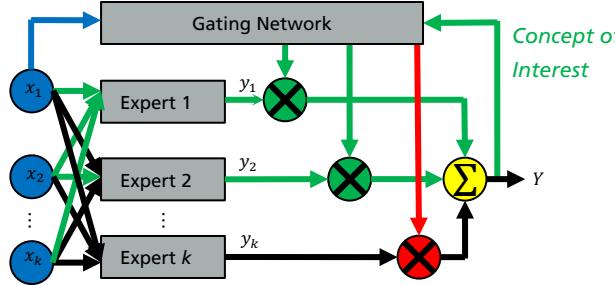


Mixture-of-Relevant-Experts

■ Relevant Path Coding



■ Mixture-of-Experts



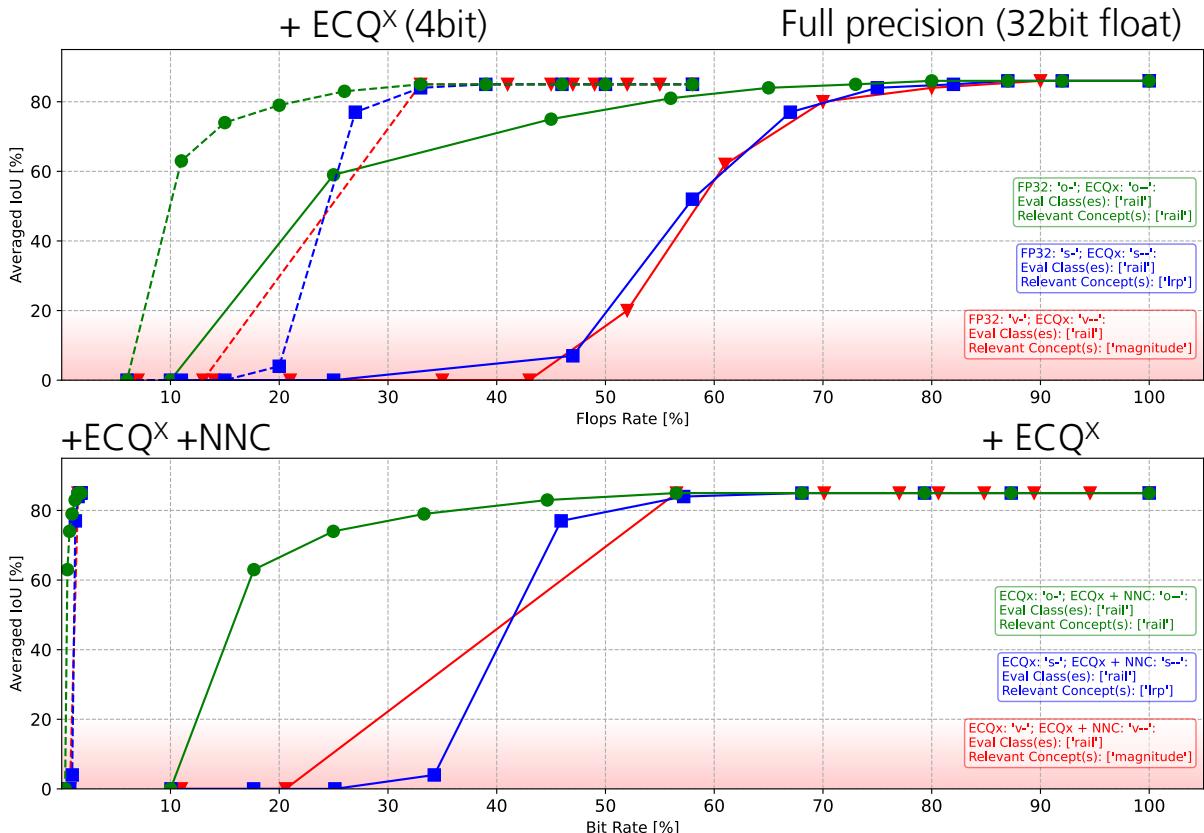
- Identifizierung konzeptspezifischer, effizienter Expertenpfade innerhalb des Supernetzes
- Gating-Mechanismus ermöglicht Variation der Expertengröße sowie Zusammenschalten mehrerer Experten (Experte für Klasse Schiene + Experte für Verkehrszeichen)
- Z.B. als Frühwarnsystem oder Domänenanpassung

Ergebnisse von Mixture-of-Relevant-Experts + ECQ^X + NNC

BerDiBa synthetisch
“rail”-Experten:

➤ >75% weniger
Rechenkomplexität

➤ >97% Kompression



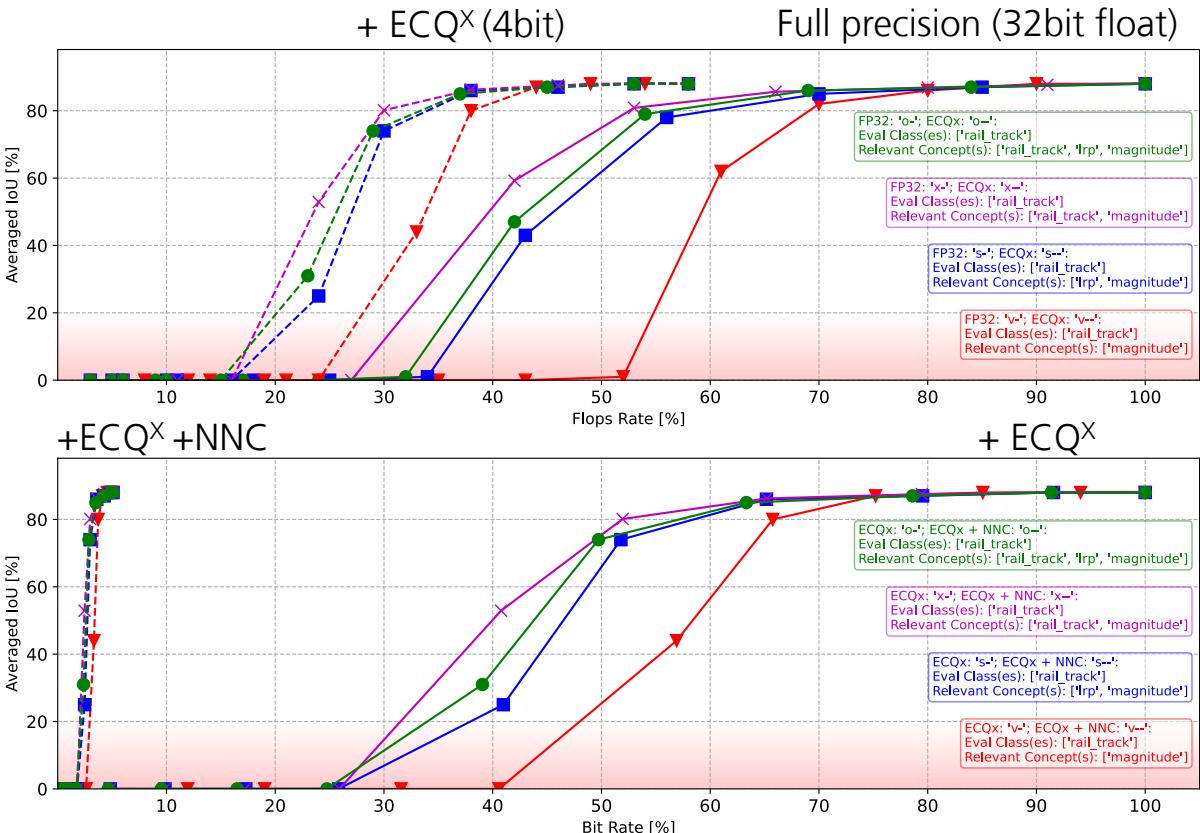
Ergebnisse von Mixture-of-Relevant-Experts + ECQ^X + NNC

RailSem19

“rail_track”-Experten:

- >60% weniger Rechenkomplexität

- >95% Kompression



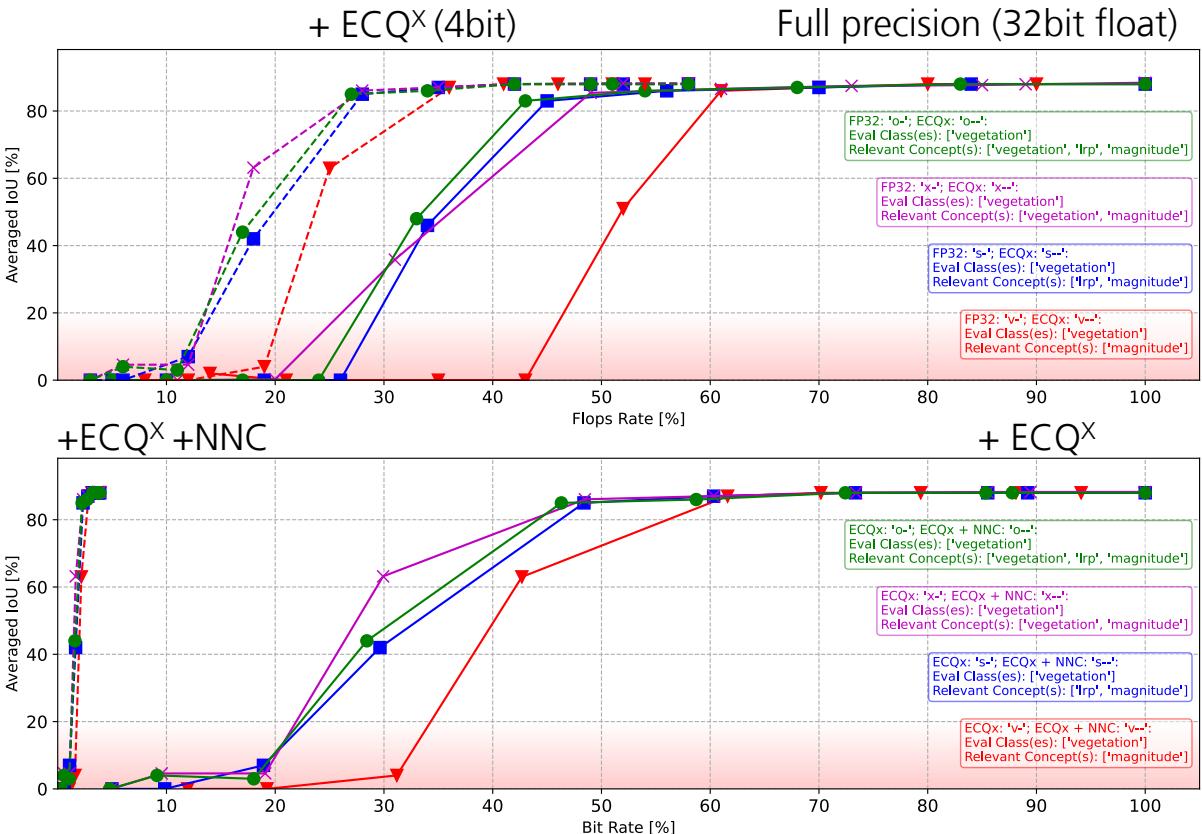
Ergebnisse von Mixture-of-Relevant-Experts + ECQ^X + NNC

CityScapes

“vegetation”-Experten:

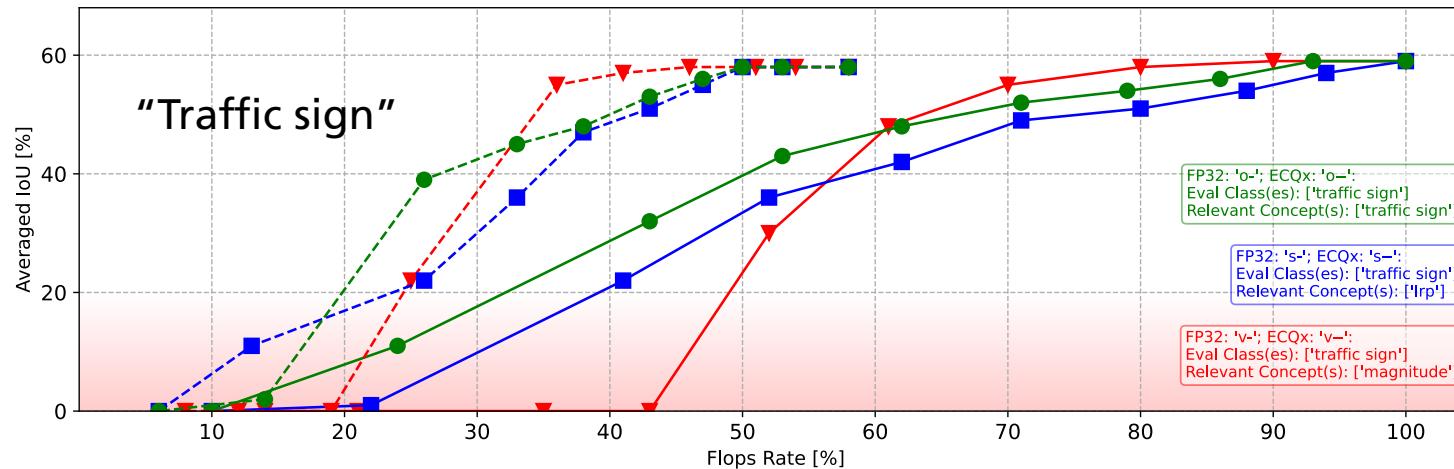
- >70% weniger Rechenkomplexität

- >95% Kompression



Limitierungen der Mixture-of-Relevant-Experts

- Relevanz-basiertes Gating ist nicht bei allen Klassen / Konzepten dem rein Gewichtswert-basierten Gating überlegen:
 - Insbesondere bei Klassen, die mit vergleichsweise wenigen Pixeln in den Trainingsdaten repräsentiert sind (z.B. kleine, seltene, vielgestaltige Objektklassen)



BerDiBa XAI-Kompression Demonstrator

TERMINAL

```
/home/nico/projects/rel_path_coding/pruning/already_generated_files/evaluation_d
ata/ECQX_MODE/stats_dict_railsem19.pt
```

Relevant Path Configuration applied to Running Model:
 Eval Class(es):['traffic_light', 'traffic_sign', 'human', 'car'],
 Rel Paths Class(es):['most_critical', 'average_rel_path', 'magnitude_path']

Relevant Paths running at parameter-rate=61% w.r.t. to the Super-Model

Relevant Paths running at parameter-rate=48% w.r.t. to the Super-Model

Relevant Paths running at parameter-rate=61% w.r.t. to the Super-Model

Enter your command here! For help, click the help-button on the top right!

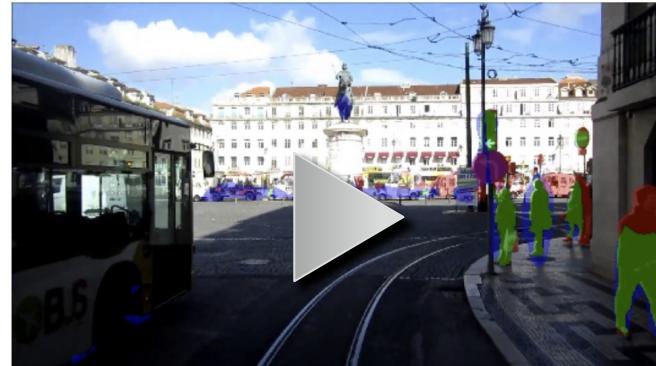
Configuration Step 1: RELEVANT PATHS

1.1 Which precision should your trainable parameters have?

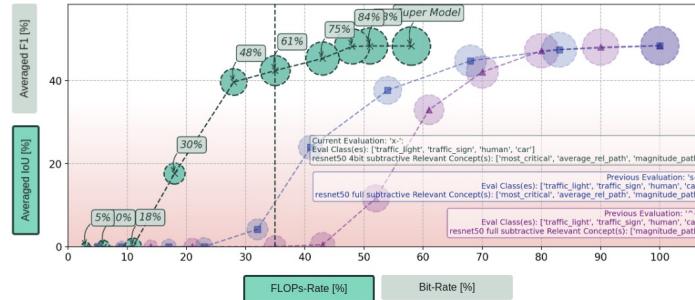
full_precision 4_bit

1.2 For which classes do you want to run the respective most relevant paths in your super-model?

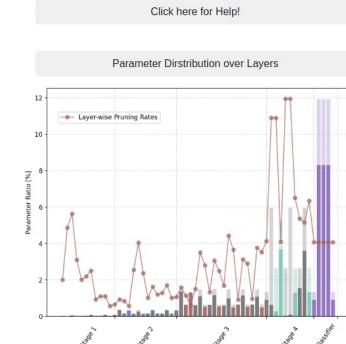
- 0: road 1: sidewalk 2: construction
- 3: tram_track 4: fence 5: pole
- 6: traffic_light 7: traffic_sign 8: vegetation
- 9: terrain 10: sky 11: human
- 12: rail_track 13: car 14: truck
- 15: trackbed 16: on_rails 17: rail_raised
- 18: rail_embedded 19: traffic_light, traffic_sign, human, car
- 20: average_rel_path
- 21: magnitude_path



◀ ▶ correct false_positive false_negative



How many of the most relevant paths of the selected classes do you want to utilize in super-model?



Path Similarity Heatmap

Layer	Path Similarity	Pruning Rate [%]	Parameter Score [%]
backbone.conv1	86.36	12.5	0.02
backbone.layer1.0.conv1	81.97	37.11	0.01
backbone.layer1.0.conv2	78.99	43.85	0.09
backbone.layer1.0.conv3	91.77	21.88	0.04
backbone.layer1.0.downsample.0	92.79	12.5	0.04
backbone.layer1.1.conv1	93.99	14.06	0.04
backbone.layer1.1.conv2	91.94	16.75	0.09
backbone.layer1.1.conv3	94.06	3.12	0.04
backbone.layer1.2.conv1	94.06	4.69	0.04
backbone.layer1.2.conv2	95.31	4.69	0.09
backbone.layer1.2.conv3	95.31	0.0	0.04
backbone.layer2.0.conv1	95.31	0.78	0.08
backbone.layer2.0.conv2	96.48	3.11	0.37
backbone.layer2.0.conv3	97.03	2.34	0.17
backbone.layer2.0.downsample.0	96.35	0.0	0.33
backbone.layer2.1.conv1	95.16	17.19	0.17
backbone.layer2.1.conv2	87.89	30.13	0.37
backbone.layer2.1.conv3	95.62	15.62	0.17
backbone.layer2.2.conv1	96.56	3.91	0.17
backbone.layer2.2.conv2	93.36	9.16	0.37
backbone.layer2.2.conv3	96.41	5.47	0.17
backbone.layer2.3.conv1	95.0	6.25	0.17
backbone.layer2.3.conv2	91.41	9.91	0.37

Zusammenfassung

- Die aus Phase 1 als beste Kandidaten identifizierten Kompressionsmethoden wurden optimiert und um eine Schnittstelle für Erklärbarkeitsmethoden (XAI) erweitert.
- Die XAI Methoden *LRP* und *CRP* wurden für die BerDiBa Modelle angepasst und mittels neuer Lernregeln geeignete Relevanzmetriken erzeugt.
Diese wurden als Kriterium bei der Parameterreduktion (*Mixture-of-Relevant-Experts*) und Präzisionsreduktion (*ECQ^X*) so eingesetzt, dass die Effizienzsteigerung der Modelle weiter verbessert werden konnte.
- Die Konzept-spezifischen Ergebnisse der optimierten XAI-Kompressionspipeline wurden in einem interaktiven Demonstrator veranschaulicht.

Fraunhofer-Institut für Nachrichtentechnik, Heinrich-Hertz-Institut, HHI

**THANK YOU FOR
YOUR ATTENTION.**

Contact:

--

Daniel Becking

daniel.becking@hhi.fraunhofer.de

+49 30 31002-406

Einsteinufer 37

10587 Berlin



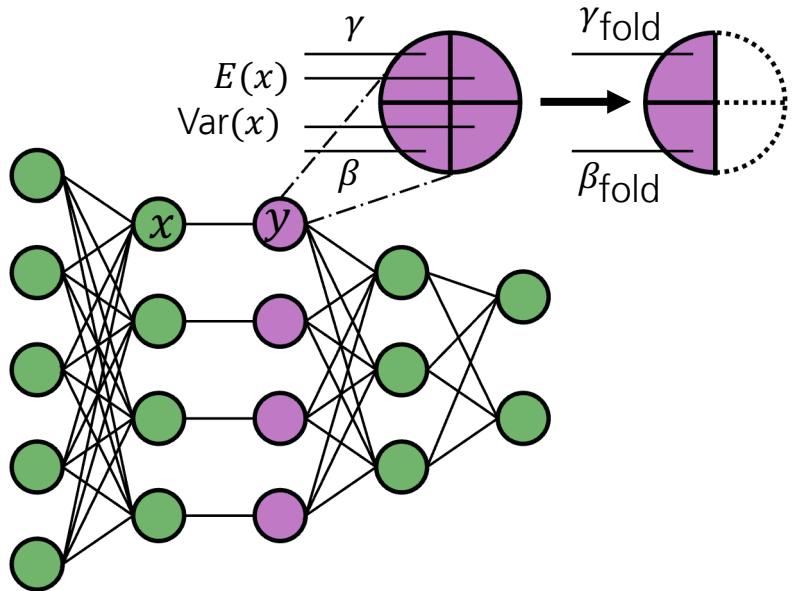
Literaturverzeichnis

- [1] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff und H. Adam, „Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018.
- [2] O. Zendel, M. Z. M. Murschitz, D. Steininger, S. Abbasi und C. Beleznai, „RailSem19: A Dataset for Semantic Rail Scene Understanding,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2019.
- [3] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth und B. Schiele, „The cityscapes dataset for semantic urban scene understanding,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.
- [4] D. Becking, P. Haase, H. Kirchhoffer, K. Müller und W. Samek, „NNCodec: An Open Source Software Implementation of the Neural Network Coding ISO/IEC Standard,” in ICML 2023 Workshop Neural Compression: From Information Theory to Applications, 2023.
- [5] S. Wiedemann, S. Shivapakash, D. Becking, P. Wiedemann, W. Samek, F. Gerfers und T. Wiegand, „FantastIC4: A hardware-software co-design approach for efficiently running 4bit-compact multilayer perceptrons,” IEEE Open Journal of Circuits and Systems, Bd. 2, pp. 407-419, 2021.
- [6] D. Becking, M. Dreyer, W. Samek, K. Müller und S. Lapuschkin, „ECQx: Explainability-Driven Quantization for Low-Bit and Sparse DNNs,” in Beyond Explainable AI, Lecture Notes in Computer Science Volume 13200, Springer, 2022, pp. 271-296.
- [7] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller und W. Samek, „On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation,” PLoS ONE, Public Library of Science San Francisco, Bd. 10, Nr. 7, 2015.
- [8] R. Achitbat, M. Dreyer, I. Eisenbraun, S. Bosse, T. Wiegand, W. Samek und S. Lapuschkin, „From attribution maps to human-understandable explanations through concept relevance propagation,” Nature Machine Intelligence, Bd. 5, Nr. 9, pp. 1006-1019, 2023.

Supplementary Materials



Batchnorm Folding



$$y = \frac{x - E(x)}{\sqrt{\text{Var}(x) + \epsilon}} \cdot \gamma + \beta$$

$$\epsilon = 1e - 5$$

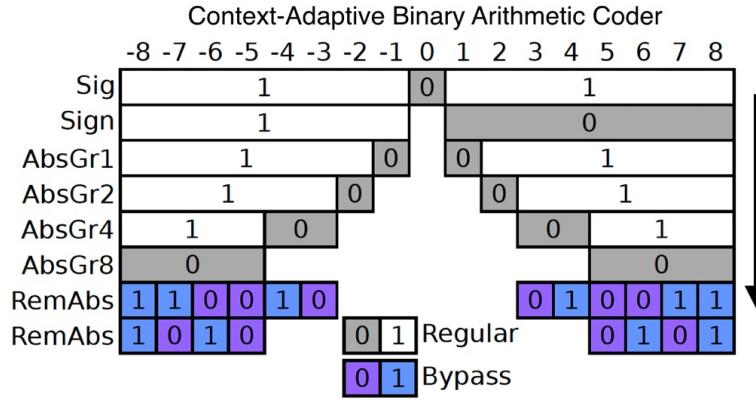
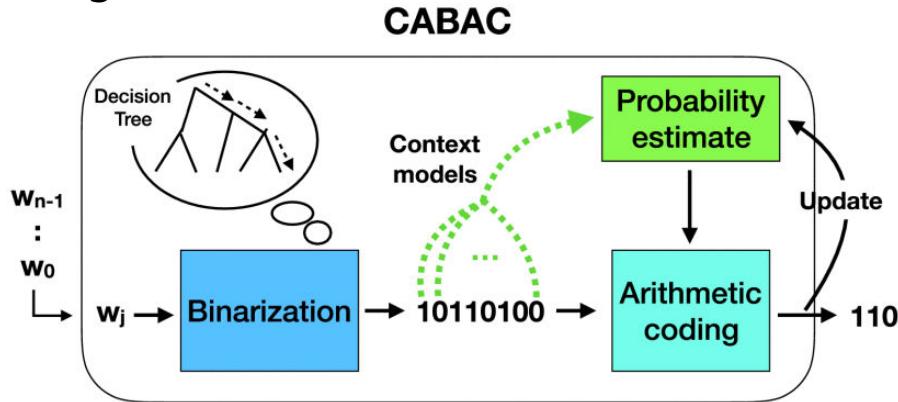
$$\gamma_{\text{fold}} = \frac{\gamma}{\sqrt{\text{Var}(x) + \epsilon}}$$

$$\beta_{\text{fold}} = \beta - (\gamma_{\text{fold}} \cdot E(x))$$

At the decoder, γ_{fold} and β_{fold} replace γ and β in the unfolded structure, whereas the elements of Var and E are set to 1 and 0.

DeepCABAC

- Entropy coding in NNC
- Highly efficient coding of quantized neural network weights.
- Context modelling is updated on-the-fly and allows to quickly adapt to all types of local weights statistics.



Examples

$1 \rightarrow 100$

$-4 \rightarrow 111101$

$7 \rightarrow 10111010$

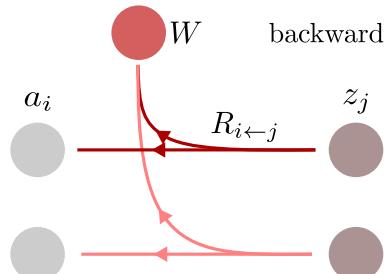
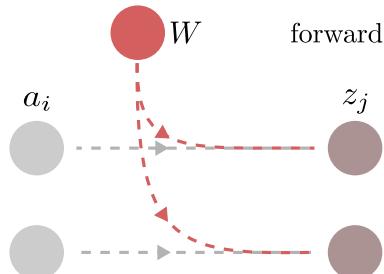
Inferring relevances from feature space to weight space

- LRP-relevances per convolutional and FC weight:

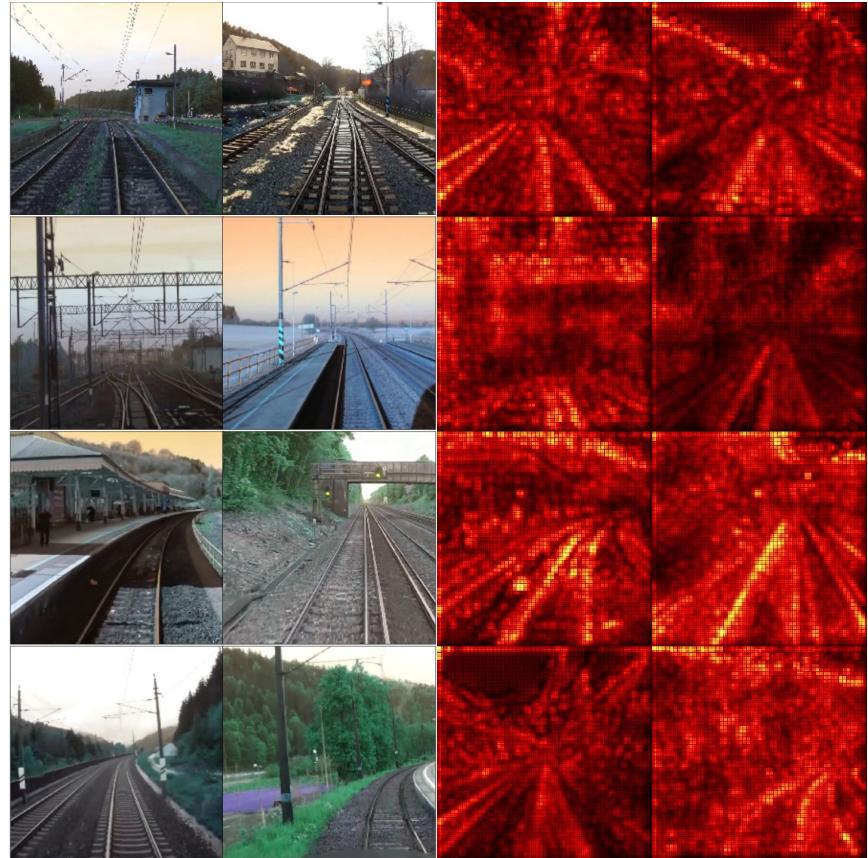
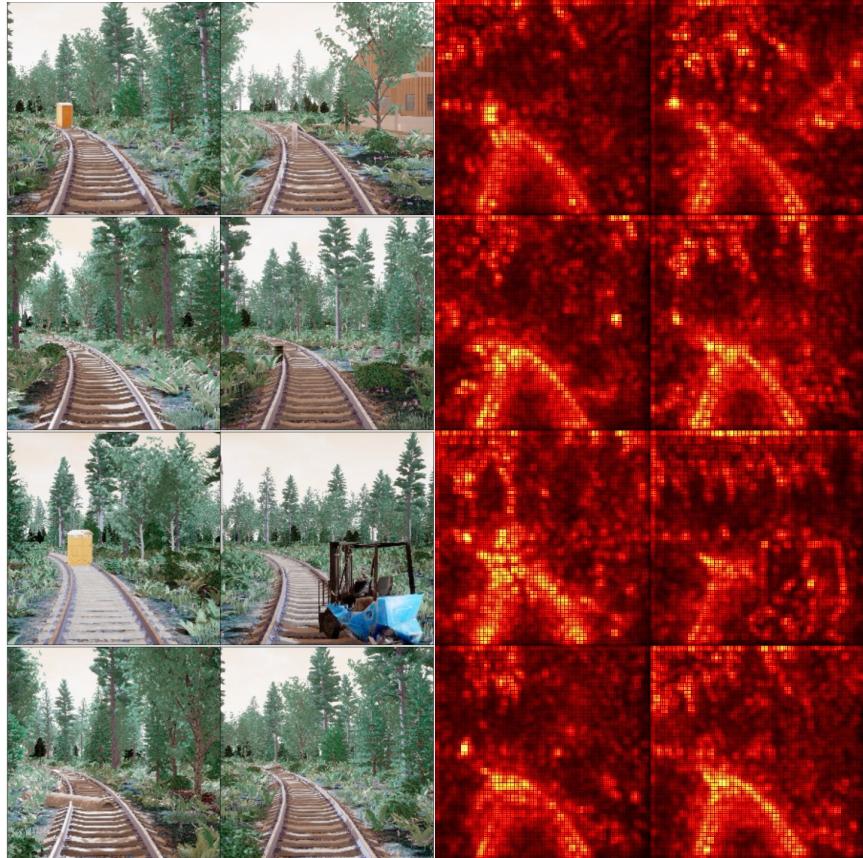
“Collection” of Relevance $R_i = \sum_j R_{i \leftarrow j}$

Contribution by weight
 $R_{i \leftarrow j} = \frac{z_{ij}}{z_j} R_j = a_i w_{ij} \frac{R_j}{z_j}$

Idea: Weights can also receive relevance – not only neurons

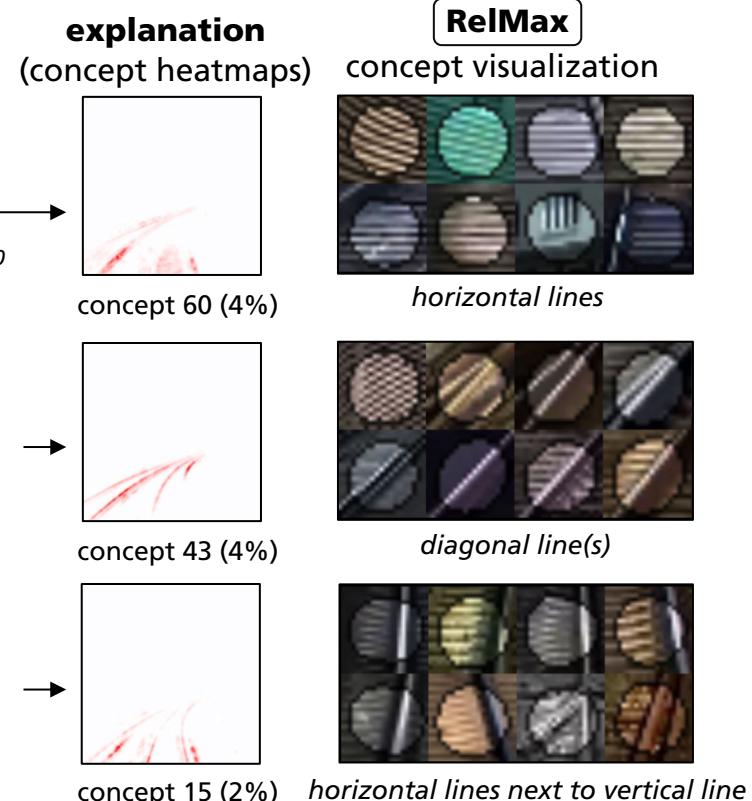
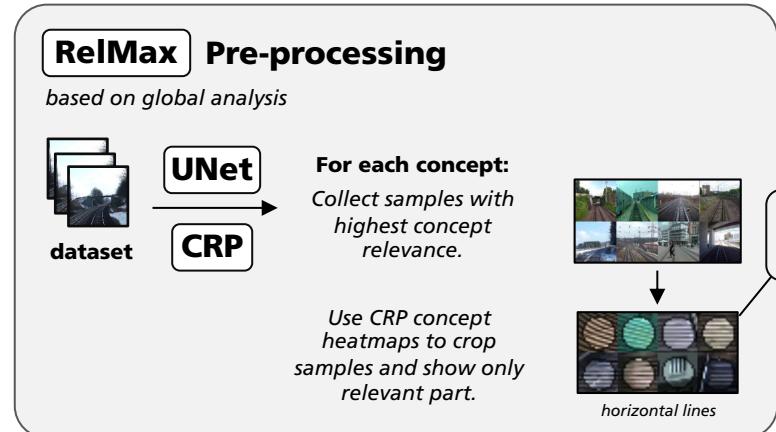
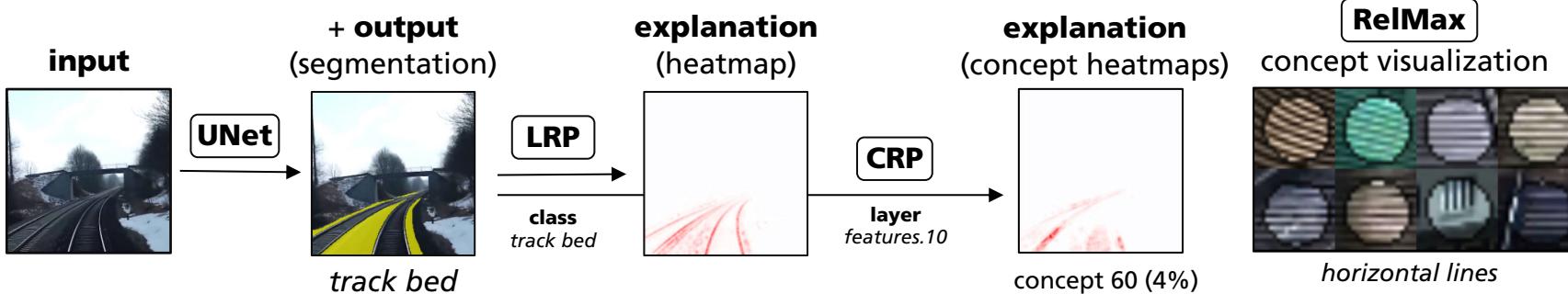


LRP Relevance heatmaps for synthetic data and RailSem19



CRP & RelMax for Concept Visualization

methodology published in [8]



Dataset 1: BerDiBa Synthetic Data

- 5000 RGB images as input plus the according segmentation maps as target.
- 7 classes encoded in 7 RGB triplets: rail, trackbed, vegetation, anomaly on track, ground, buildings, sky.
- Findings while implementing dataloaders: some pixels are of undefined classes.



Dataset 2: RailSem19^[2]

- 8500 images taken from the ego-perspective of a rail vehicle (trains and trams)
- 19 classes: road, sidewalk, construction, tram track, fence, pole, traffic light, traffic sign, vegetation, terrain, sky, human, rail track, car, truck, trackbed, on-rails, rail raised, rail embedded, void



[2] O. Zendel, M. Murschitz, M. Zeilinger, D. Steininger, S. Abbasi and C. Beleznai, "RailSem19: A Dataset for Semantic Rail Scene Understanding," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019, pp. 1221-1229.

Dataset 3: CityScapes^[3]

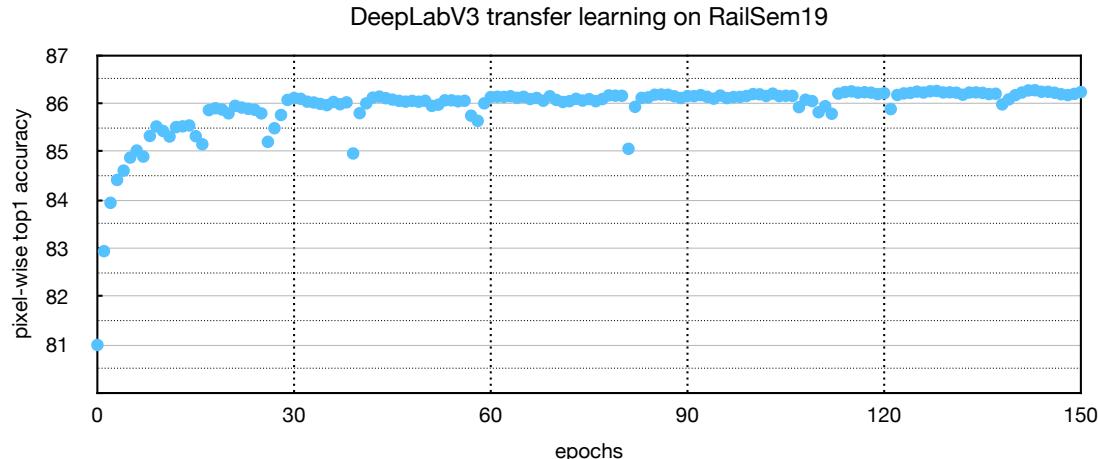
- RailSem19 compatible labels, includes video sequences
- Recorded in street scenes from 50 different cities, pixel-level annotations of 5 000 frames in addition to a larger set of 20 000 weakly annotated frames.



[3] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth und B. Schiele, „The cityscapes dataset for semantic urban scene understanding,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016

Transfer-learning the model to rail data

- Applying gained knowledge to a different but related problem:
 - DeepLabV3 with ResNet101 backbone^[1], pre-trained[‡] on COCO val'17 (21 classes)
 - Updating the classifier module to predict 7 (BerDiBa synth.) or 19 (RailSem19) instead of 21 classes
 - 80:20 train:validation data,
cross entropy loss,
ADAM optimization,
initial learning rate 1e-04
 - Pixel-wise Top1 acc:
RailSem19: 86.2%
BerDiBa synth.: 91.2%



[‡] https://pytorch.org/hub/pytorch_vision_deeplabv3_resnet101/