CNN Pre-Training Using Convolutional Autoencoders

Russell Hofmann, Sabbir Ahmed, Maximilian Kohlbrenner

TU Berlin

Setup

Experimental setup

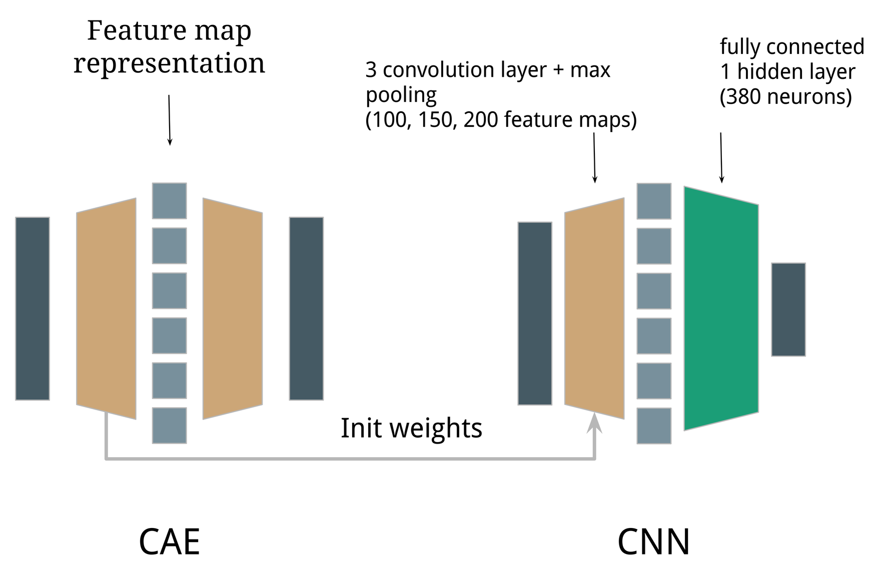


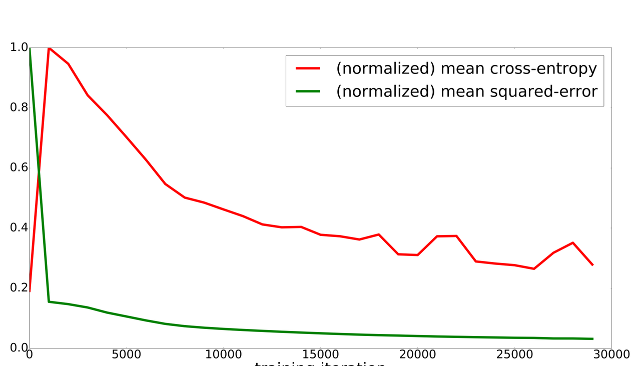
Fig. 1: Pre-training setup

Convolutional Autoencoders

- Convolutional architecture, reconstruction similar to deconvolution method
- Regularization achieved using max-pooling (Masci, J., Meier, U., Cireşan, D. and Schmidhuber, J., 2011)

Fig. 2: Input (first row), and a selection of feature maps from the subsequent convolutional layers (rows 2,3 and 4). The last row (feature maps for the 3rd convolution layer) makes up the hidden representation of the CAE

CAE: Activation and Error Function



- Activation function: scaled-tanh (saturating, sigmoidal)
- Error function: mean-squared error (mean cross-entropy produced blank reconstruction)

Pre-Training Experiments

MNIST

Fig. 3: *MNIST* example images (top) and CAE reconstructions (bottom)

Dataset Characteristics:

- Comparatively simple (linear classifier achieves > 90% accuracy)

Experimental results:

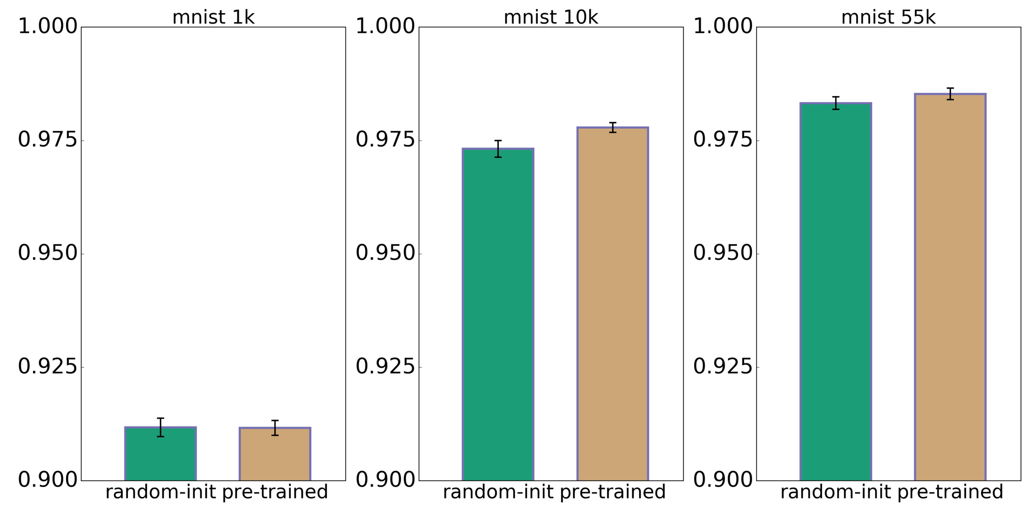


Fig. 6: *MNIST*: test set accuracy comparison for different training set sizes (stddev in black)

CIFAR-10

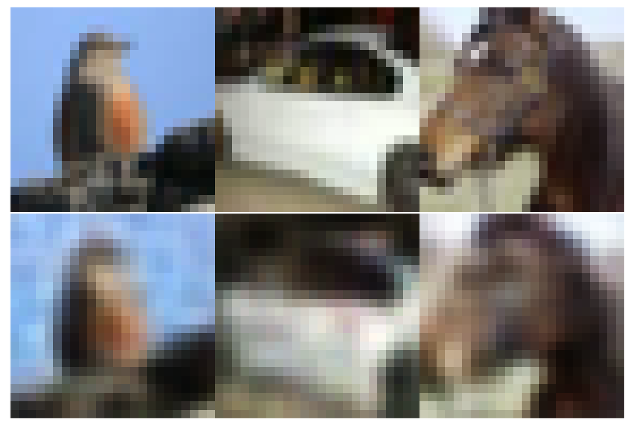


Fig. 4: *CIFAR-10* example images(top) and CAE reconstructions(bottom)

Dataset Characteristics:

- natural image classification (difficult)
- comparably low resolution

Experimental results:

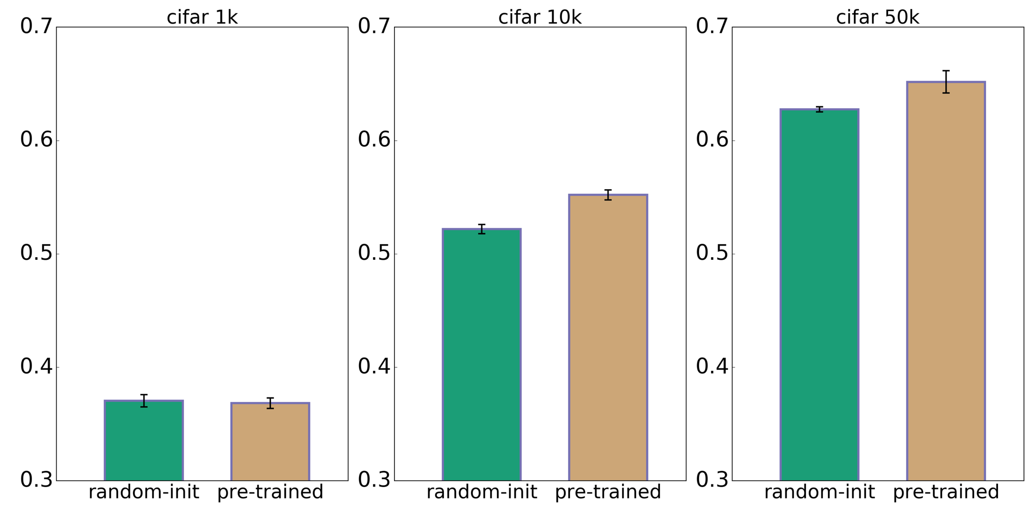


Fig. 7: *CIFAR-10*: test set accuracy comparison for different training set sizes (stddev in black)

Extended Cohn-Kanade (CK+)

Fig. 5: CK+ input image (top) and reconstruction (bottom)

Dataset Characteristics:

- pre-processing and set splitting
- very few data (696 train, 87 validation, 198 test in our case)

Experimental results:

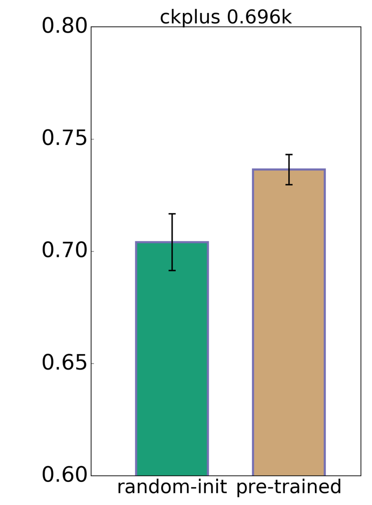


Fig. 8: *CK+*: test set accuracy comparison for 696 training images (stddev in black)

Conclusions

- Pre-Training consistently increases accuracy over all tested datasets and most training sizes (figures 6, 7 and 8)
- After training, pre-trained network's first-layer filters show a clearer structure (figure 9)
- CAE error function is crucial
- When changing from sigmoidal to ReLU activation function, max-pooling no longer seems sufficient for CAE regularization (figure 10)

Fig. 9: ***CIFAR-10* CNN** selection of first layer filters. pre-trained (left), random-init (right)

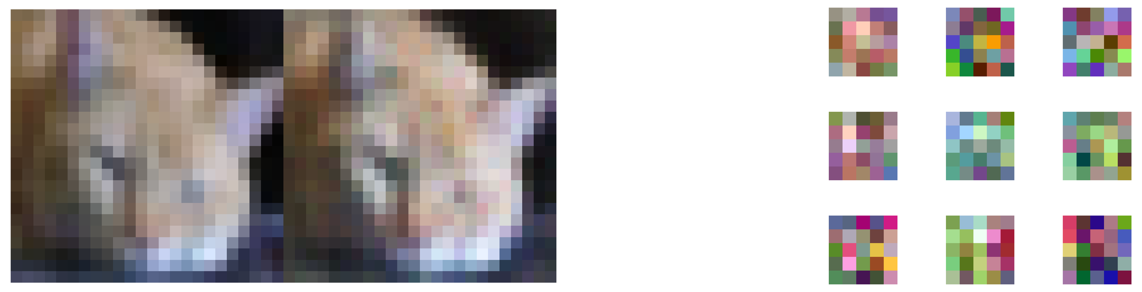


Fig. 10: **ReLU CAE** example input + reconstruction (left), selection of first layer filters (right)