8. Supplementary Material

8.1. Details of Experimental Setups

The following paragraphs describe the precise experimental settings used to obtain results in this paper. The source code for reproducing the results on Penn Treebank, enwik8 and text8 experiments is available at https://github.com/julian121266/RecurrentHighwayNetworks on Github.

Optimization

In these experiments, we compare RHNs to Deep Transition RNNs (DT-RNNs) and Deep Transition RNNs with Skip connections (DT(S)-RNNs) introduced by Pascanu et al. (2013). We ran 60 random hyperparamter settings for each architecture and depth. The number of units in each layer of the recurrence was fixed to $\{1.5 \times 10^5, 3 \times 10^5, 6 \times 10^5, 9 \times 10^5\}$ for recurrence depths of 1, 2, 4 and 6, respectively. The batch size was set to 32 and training for a maximum of 1000 epochs was performed, stopping earlier if the loss did not improve for 100 epochs. $tanh(\cdot)$ was used as the activation function for the nonlinear layers. For the random search, the initial transform gate bias was sampled from $\{0, -1, -2, -3\}$ and the initial learning rate was sampled uniformly (on logarithmic scale) from $[10^0, 10^{-4}]$. Finally, all weights were initialized using a Gaussian distribution with standard deviation sampled uniformly (on logarithmic scale) from $[10^{-2}, 10^{-8}]$. For these experiments, optimization was performed using stochastic gradient descent with momentum, where momentum was set to 0.9.

Penn Treebank

The Penn Treebank text corpus (Marcus et al., 1993) is a comparatively small standard benchmark in language modeling. The and pre-processing of the data was same as that used by Gal (2015) and our code is based on Gal's (Gal, 2015) extension of Zaremba's (Zaremba et al., 2014) implementation. To study the influence of recurrence depth, we trained and compared RHNs with 1 layer and recurrence depth of from 1 to 10. with a total budget of 32 M parameters. This leads to RHN with hidden state sizes ranging from 1275 to 830 units. Batch size was fixed to 20, sequence length for truncated backpropagation to 35, learning rate to 0.2, learning rate decay to 1.02 starting at 20 epochs, weight decay to 1e-7 and maximum gradient norm to 10. Dropout rates were chosen to be 0.75 for the embedding layer, 0.25 for the input to the gates, 0.25 for the hidden units and 0.75 for the output activations. All weights were initialized from a uniform distribution between $[-0.04, 0.04]$. For the best 10-layer model obtained, lowering the weight decay to 1e-9 further improved results.

Enwik8

The Wikipedia enwik8 dataset (Hutter, 2012) was split into training/validation/test splits of 90 M, 5 M and 5 M characters similar to other recent work. We trained three different RHNs. One with 5 stacked layers in the recurrent state transition with 1500 units, resulting in a network with ≈23.4 M parameters. A second with 10 stacked layers in the recurrence with 1000 units with a total of ≈20.1 M parameters and a third with 10 stacked layers and 1500 units with a total of of ≈46.0 M parameters. An initial learning rate of 0.2 and a learning rate decay of 1.04 after 5 epochs was used. Only the large model with 10 stacked layers and 1500 units used a learning rate decay of 1.03 to ensure for a proper convergence. Training was performed on mini-batches of 128 sequences of length 50 with a weight decay of 0 for the first model and 1e-7 for the other two. The activation of the previous sequence was kept to enable learning of very long-term dependencies (Graves, 2013). To regularize, variational dropout (Gal, 2015) was used. The first and second model used dropout probabilities of 0.1 at input embedding, 0.3 at the output layer and input to the RHN and 0.05 for the hidden units of the RHN. The larger third model used dropout probabilities of 0.1 at input embedding, 0.4 at the output layer and input to the RHN and 0.1 for the hidden units of the RHN. Weights were initialized uniformly from the range [-0.04, 0.04] and an initial bias of -4 was set for the transform gate to facilitate learning early in training. Similar to the Penn Treebank experiments, the gradients were re-scaled to a norm of 10 whenever this value was exceeded. The embedding size was set to 205 and weight-tying (Press & Wolf, 2016) was not used.

Text8

The Wikipedia text8 dataset (Hutter, 2012) was split into training/validation/test splits of 90 M, 5 M and 5 M characters similar to other recent work. We trained two RHNs with 10 stacked layers in the recurrent state transition. One with 1000 units and one with 1500 units, resulting in networks with ≈20.1 M and ≈45.2 M parameters, respectively. An initial learning rate of 0.2 and a learning rate decay of 1.04 for the 1000 unit model and 1.03 for the 1500 units model was used after 5 epochs. Training was performed on mini-batches of 128 sequences of length 100 for the model with 1000 units and 50 for the model with 1500 units with a weight decay of 1e-7. The activation of the previous sequence was kept to enable learning of very long-term dependencies (Graves, 2013). To regularize, variational dropout (Gal, 2015) was used with dropout probabilities of 0.05 at the input embedding, 0.3 at the output layer and input to the RHN and 0.05 for the hidden units of the RHN for the model with 1000 units. The model with 1500 units used dropout probabilities of 0.1 at the input embedding, 0.4 at the output layer and at the input to the RHN and finally 0.1 for the dropout probabilities of the hidden units of the RHN. Weights were initialized uniformly from the range [-0.04, 0.04] and an initial bias of -4 was set for the transform gate to facilitate learning early in training. Similar to the Penn Treebank experiments, the gradients were rescaled to a norm of 10 whenever this value was exceeded. The embedding size was set to 27 and weight-tying (Press & Wolf, 2016) was not used.

Lesioning Experiment Figure 6 shows the results of the lesioning experiment from section 6. This experiment was conducted on the RHN with recurrence depth 6 trained on the JSB Chorales dataset as part of the Optimization experiment in subsection 5.1. The dashed line corresponds to the training error without any lesioning. The x-axis denotes the index of the lesioned highway layer and the y-axis denotes the log likelihood of the network predictions.

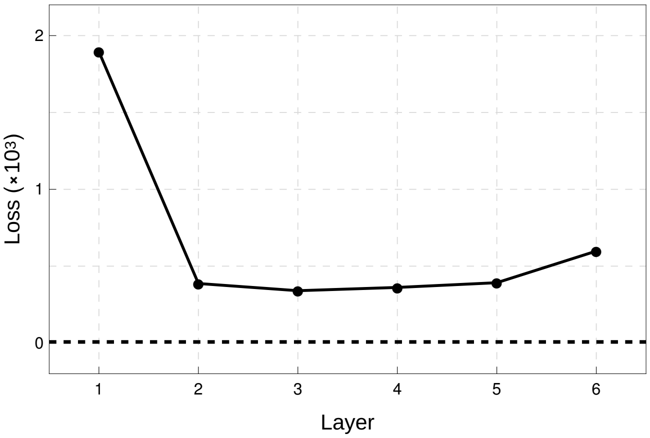

Figure 6: Changes in loss when the recurrence layers are biased towards carry behavior (effectively removed), one layer at a time.