Appendices

A Implementation Details

Our implementation is based on TensorFlow.[14] For both experiments, we use the similar implementation strategies for the baselines and our model, aiming for a fair comparison.

A.1 Text Summarization

We train the models using Adam (Kingma and Ba, 2015) with a batch size of 64. We use the default values in TensorFlow Adam implementation for initial learning rate η_0, β_1, β_2, and ε. The models are trained for up to 20 epochs, with the learning rate annealed at a rate of 0.2 every 4 epochs. A weight decay of $0.01 \times \eta$ is applied to all parameters, with η being the current learning rate. The ℓ_2-norms of gradients are clipped to 1.0. Early stopping is applied based on ROUGE-L performance on the development set.

The weights of the output $\mathrm{softmax}$ function are tied with the word embeddings, which are randomly initialized. For the encoders, we use 256-dimensional 3-layer BiLSTMs in the Gigaword experiment, and 300-dimensional 2-layer BiLSTMs in the NYT experiment, both with residual connections (He et al., 2015); the decoders are one-layer (adaptive) LSTMs, and have the same size as the encoders, and so do the word embeddings. We apply variational dropout (Kingma et al., 2015) in the encoder RNNs, and dropout (Srivastava et al., 2014) in the embeddings and the $\mathrm{softmax}$ layer, the rates of which are empirically selected from $[0.15, 0.25, 0.35]$. The last hidden state at the top layer of the encoder is fed through an one-layer $\tanh$-MLP, and then used to as the decoder's initial state. We use the attention function by Luong et al. (2015), and copy mechanism by See et al. (2017). The exemplar encoder (§3.2) uses one-layer 32/50 BiLSTM for Gigaword and NYT experiments, respectively. For numerical stability, $\boldsymbol{\lambda}$ (Equation 11) is scaled to have ℓ_2 norms of $\sqrt{d}$, with d being the hidden size of the adaptive decoder (Peng et al., 2018b).

In the Gigaword experiment, we use 25K BPE types, and limit the maximum decoding length to be 50 subword units; while for NYT, we use 10K types with a maximum decoding length of 300, and further truncate the source to the first 1000 units. During evaluation, we apply beam search of width 5, with a 1.0 length penalty (Wu et al., 2016).

A.2 Data-to-text Generation

We in general follow the implementation details in the summarization experiment, with the following modifications:

- We do *not* apply byte-paired encoding here, and use a vocabulary of size 50K.
- The word embeddings are initialized using 840B version 300-dimensional `GloVe` (Pennington et al., 2014) and fixed during training. Further, the $\mathrm{softmax}$ weights are *not* tied to the embeddings.
- Three-layer 300-dimensional BiLSTM encoders are used, with residual connections; the exemplar encoder uses an one-layer 50-dimensional BiLSTM.
- Early stopping is applied based on development set ROUGE-4 performance.
- A maximum decoding length of 40 tokens is used.

[14] https://www.tensorflow.org/