## A Implementation Details

For all experiments, our model has 256-dimensional hidden states and 128-dimensional word embeddings. Since the pointer-generator model has the ability to deal with the OOV words, we choose a small vocabulary size of 5k, and we train the word embedding from scratch. We also truncate both the input and output sentences to 20 tokens.

For the training part, we first pre-train a pointer generator model using MLE, and then fine-tune this model with the REINFORCE, DAGGER and other variant learning algorithms respectively. In the **pre-training** phase, we use the Adagrad optimizer with learning rate 0.15 and an initial accumulator value of 0.1; use gradient clipping with a maximum gradient norm of 2; and do auto-evaluation on the validation set every 1000 iterations, in order to save the best model with the lowest validation loss. In the **fine-tuning** phase, we use the Adam optimizer with learning rate $10^{-5}$; use gradient clipping with the same setting in pre-training; and do auto-evaluation on the validation set every 10 iterations.

When applying the REINFORCE and its variant algorithms, we compute the reward as follows:

$$r(\tilde{\boldsymbol{y}}_n, \boldsymbol{y}) = \text{ROUGE-2}(\tilde{\boldsymbol{y}}_n, \boldsymbol{y}) - \frac{1}{N}\sum_{n=1}^{N}\text{ROUGE-2}(\tilde{\boldsymbol{y}}_n, \boldsymbol{y}) \quad (5)$$

where $N$ is the total number of sentences generated by random sampling, in all the experiments, we set $N = 4$.

At test time, we use beam search with beam size 8 to generate the paraphrase sentence.

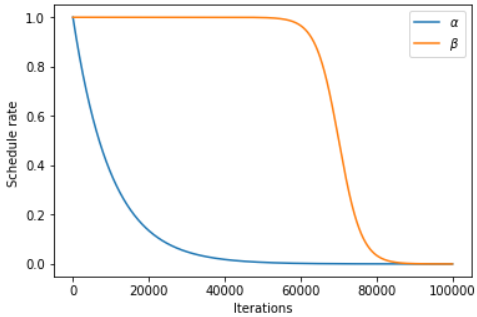


Figure 1: The schedule sampling rate for $\alpha$ and $\beta$

According to Bengio et al. (2015), we define the schedule rate $\alpha^{(i)} = k^i$ (where $0 < k < 1$, $i$ is the $i$th training iteration), and $\beta^{(i)} = k/(k + exp(i/k))$ (where $k > 1$, $i$ is the $i$th training iteration). In the experiments shown in Tabel 1, for the schedule rate $\alpha$, we set $k = 0.9999$; for the schedule rate $\beta$, we set $k = 3000$, and the resulting schedule rate curve is shown in Figure 1.

## B Additional Results

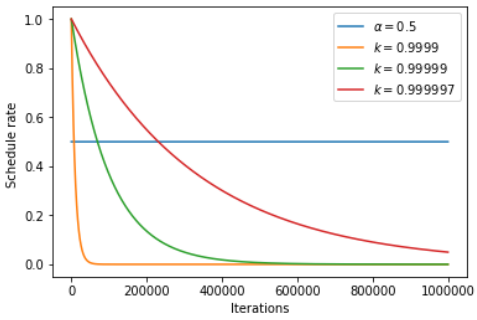


Figure 2: The schedule rate $\alpha$ in DAGGER

We try different schedule rate settings in DAGGER as shown in Figure 2, and compare their model performances on both datasets. The corresponding experimental results are shown in Table 3 and Table 4 respectively.

We find that if $\alpha$ decreases faster ($k = 0.9999$) than the model convergence speed, the model will stop improving before it learns the optimal policy; if $\alpha$ decreases slower ($k = 0.999997$) than the model convergence speed, the model will get stuck in the sub-optimal policy.

For the Quora dataset, we find our model learns the optimal policy when it gets half chance to take the ground truth word $y_{t-1}$ as $\tilde{y}_{t-1}$ (i.e. the schedule rate setting is $\alpha = 0.5$ and $\beta = 1$).

For the twitter dataset, we find our model learns the optimal policy when it has higher probability to take the decoded word $\hat{y}_{t-1}$ as $\tilde{y}_{t-1}$ (i.e. the schedule rate setting is $\alpha = 0.2$ and $\beta = 1$).

| | Schedule Rate | | | Evaluation Metrics | | | |
|---|---|---|---|---|---|---|---|
| | $\alpha$ | $\beta$ | $k_\alpha$ | ROUGE-1 | ROUGE-2 | BLEU | Avg-Score |
| 1 | $\alpha = 0.5$ | $\beta = 1$ | $k_\alpha = 1$ | **68.34** | **49.99** | **55.75** | **58.02** |
| 2 | $\alpha \to 0$ | $\beta = 1$ | $k_\alpha = 0.9999$ | 67.64 | 48.96 | 55.06 | 57.22 |
| 4 | $\alpha \to 0$ | $\beta = 1$ | $k_\alpha = 0.99999$ | 67.92 | 49.45 | 55.44 | 57.60 |
| 4 | $\alpha \to 0$ | $\beta = 1$ | $k_\alpha = 0.999997$ | 67.73 | 49.19 | 55.65 | 57.52 |

Table 3: Experiment results for DAGGER under different schedule rate settings on Quora dataset

| | Schedule Rate | | | Evaluation Metrics | | | |
|---|---|---|---|---|---|---|---|
| | $\alpha$ | $\beta$ | $k_\alpha$ | ROUGE-1 | ROUGE-2 | BLEU | Avg-Score |
| 1 | $\alpha = 0.2$ | $\beta = 1$ | $k_\alpha = 1$ | **58.95** | **44.34** | **39.04** | **47.44** |
| 2 | $\alpha \to 0$ | $\beta = 1$ | $k_\alpha = 0.9999$ | 58.84 | 44.24 | 38.95 | 47.34 |
| 4 | $\alpha \to 0$ | $\beta = 1$ | $k_\alpha = 0.99999$ | 58.81 | 44.08 | 38.85 | 47.24 |
| 4 | $\alpha \to 0$ | $\beta = 1$ | $k_\alpha = 0.999997$ | 58.79 | 44.22 | 38.88 | 47.29 |

Table 4: Experiment results for DAGGER under different schedule rate settings on Twitter dataset