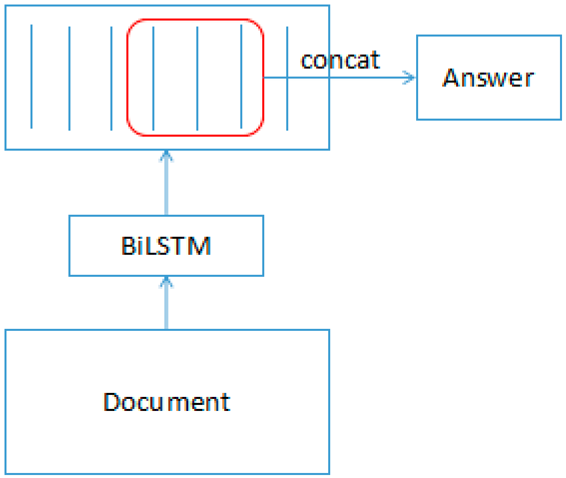
Machine Comprehension by Text-to-Text Neural Question Generation

Xingdi Yuan[1,†], Tong Wang[1,†], Caglar Gulcehre[2,†,‡], Alessandro Sordoni[1,†], Philip Bachman[1], Sandeep Subramanian[2,‡], Saizheng Zhang[2,‡], and Adam Trischler[1]

Motivation

1. Posing approriate questions is an important aspect of information acquisition in intelligent systems.
2. Learning to ask questions may improve the ability to answer them.

Encoder



$$Embedding(Doc) -> LSTM -> (h_1, h_2, ..., h_m)$$

$$A : [Hidden; Embedding]$$

$$Hidden : (h_i, h_{i+1}, ..., h_j)\ Embedding : (a_i, a_{i+1}, ..., a_j)$$

$$A -> LSTM2 -> h_a$$

h_a:concat(the final hidden state of two directions)

Computing the inital state s_0 for the decoder:

$$\mathbf{r} = \mathbf{L}\mathbf{h}^a + \frac{1}{n}\sum_{i}^{|D|} \mathbf{h}_i^d, \quad \mathbf{s}_0 = \tanh\left(\mathbf{W}_0\mathbf{r} + \mathbf{b}_0\right)$$

Decoder(vocab softmax and pointer softmax)

$$s_t = LSTM(s_{t-1}, y_{t-1}, v_t)$$

$$e_{tj} = f(h_j^d, h^a, y_{t-1}, s_{t-1})$$
$$\alpha_{tj} = \frac{exp(e_{tj})}{\sum_{i=1}^{T} exp(e_{ij})}$$

$$c_t = \sum_{i=1}^{|D|} \alpha_{ti} h_i^d$$
$$v_t = [c_t; h^a]$$

Decoder

$e_t = g(s_t, v_t, y_{t-1})$
$o_t = softmax(W_o e_t + b_o)$
Finally,$p_\theta(y_t|y < t, D, A)$ is approximated by the full pointer-softmax $p_t \in R^{|V|+|D|}$ by concatenating o_t and α_t after both are weighted by z_t:
$p_t = [z_t o_t; (1 - z_t)\alpha_t]$
z_t is a MLP gate.

Training

Corss-Entropy

$$\mathcal{L} = -\sum_t \log p_\theta(y_t | y_{<t}, D, A),$$

Penality Term1(Encourage the model not to generate answer words in the question)

$$\mathcal{L}_s = \lambda_s \sum_t \sum_{\bar{a} \in \bar{\mathcal{A}}} p_\theta(y_t = \bar{a} | y_{<t}, D, A).$$

Penelity Term2(Increase output variety)

$$\mathcal{L}_e = \lambda_e \sum_t \mathrm{p}_t^T \log \mathrm{p}_t.$$

Policy Gradient Optimization

There is a limitation in teacher-forcing that it does not teach the model how to distribute probability mass amoing examples other than the ground-truth.

Rewards

QA reward: $R_{QA}(\hat{Y}) = F1(\hat{A}, A)$

Fluency(Language Model matrix):

$R_{PPL}(\hat{Y}) = -2^{-\frac{1}{T}\sum_{t=1}^{T} log_2 P_{LM}(\hat{y}_t)|\hat{y}<t}$

Combination

$$R_{\mathrm{PPL+QA}}(\hat{Y}) = \lambda_{\mathrm{QA}} R_{\mathrm{QA}}(\hat{Y}) + \lambda_{\mathrm{PPL}} R_{\mathrm{PPL}}(\hat{Y}),$$

Reinforce

$$\mathcal{L}_{\mathrm{RL}} = -\mathbb{E}_{\hat{Y} \sim \pi(\hat{Y}|D,A)}[R(\hat{Y})],$$

$$\nabla \mathcal{L}_{\mathrm{RL}} \approx \sum_{t=1} \nabla \log \pi(\hat{y}_t|\hat{y}_{<t}, D, A) \frac{R(\hat{Y}) - \mu_R}{\sigma_R},$$

Using beam search instead of sampling when computing the expectation.
The L_{RL} is based on beam search, and the L_{CE} is based on ground-truth.

Experiments

	NLL	BLEU	F1	QA	PPL
Seq2Seq	45.8	4.9	31.2	45.6	153.2
Our System	**35.3**	10.2	39.5	65.3	175.7
+ PG (R_{PPL})	35.7	9.2	38.2	61.1	**155.6**
+ PG (R_{QA})	39.8	**10.5**	**40.1**	**74.2**	300.9
+ PG ($R_{\text{PPL+QA}}$)	39.0	9.2	37.8	70.2	183.1
Question LM	-	-	-	-	87.7
MPCM	-	-	-	70.5	-