Structured prediction with reinforcement learning.

(2009)

Standard value-basedRL algorithm (SARSA, OLPOMDP, but not use the ground-truth for value prediction)

An investigation of imitation learning algorithms for structured prediction (2012)

Imitation learning such as SEARN, DAGGER, rely on an expert policy to provide action sequence that the policy learns to initiate. But not always east or possible to construct such policy

Sequence level training with recurrent neural networks

(2015)

Trained a translation model by gradually transitioning from maximum likelihood learning into optimising BLEU or ROUGE scores using REINFORCE (but has high variance and does not exploit the availability of the ground-truth like critic network does)

One contribution is Mixed Incremental Cross-Entropy Reinfroce(MIXER), which use incremental learning and a hybrid loss function combining REINFORCE and cross-entry.

AN ACTOR-CRITIC ALGORITHM FOR SEQUENCE PREDICTION

(2017)

Introduction:

RNN is doing great when trained to predict the next output token given the input and previous tokens.

Standard way to train RNN is maximing the log-likelihood of ‘correct’ token given a history of the previous “correct'' ones, called teacher forcing. But its conditioned on guesses, error may accumulates, especially for lone sequence.

Bengio et al. (2015) use the token k from the ground-truth answer as the target for the network at step k, whereas Ranzato et al. (2015) rely on the REINFORCE algorithm (Williams, 1992) to decide whether or not the tokens from a sampled prediction lead to a high task-specific score, such as BLEU (Papineni et al., 2002) or ROUGE (Lin & Hovy, 2003).

In this work, they Tain an additional network called critic to output the value of each token. Inspired by actor-critic approach in RL field. In their case, the reward is analogous to the task-specific score associated with a prediction. They have also use temporal difference methods to train the critic.

Model setting:

——RNN for sequence prediction, add a probabilistic layer to the output, so each hidden state depends on the previous hidden stat e and previous output. Use a soft attention mechanism

$$
\begin{aligned}
y_t &\sim g(s_{t-1}, c_{t-1}) \\
s_t &= f(s_{t-1}, c_{t-1}, e(y_t)) \\
\alpha_t &= \beta(s_t, (h_1, \ldots, h_L)) \\
c_t &= \sum_{j=1}^{L} \alpha_{t,j} h_j
\end{aligned}
$$

(h1,…,hL) is the encoded sequence of X

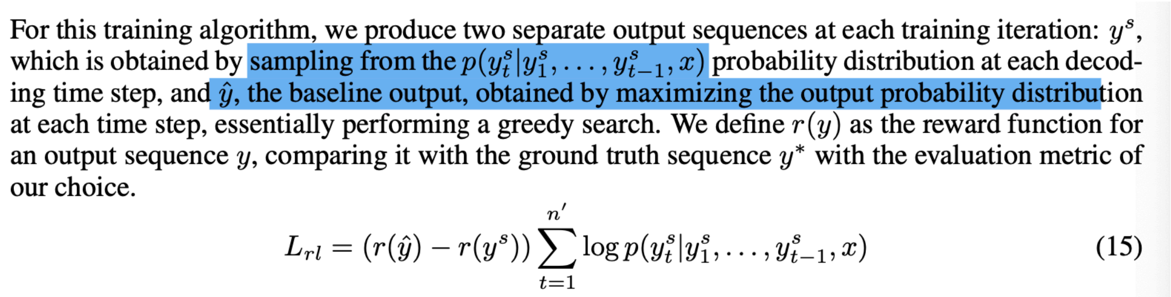
——Value function: Instead of receiving full return at the end of sequence prediction such as computing the metric of result at the end, we receive intermediate rewards along the prediction. This ease the learning for the critic, we also use reward shaping.

A DEEP REINFORCED MODEL FOR ABSTRACTIVE SUMMARIZATION (2017)

RL is a way of training an agent to interact with a given environment in order to maximise a reward. It is used usually when an agent hasty perform discrete action before obtaining a reward, or when the metric to optimise is not differentiable and traditional supervised learning methods cannot be used, such as BLEU, ROUGE, METERO.

In this work, they explore different ways of training encoder-decoder model, particularly using reinforcement learning- based algorithm.

Proposed Policy learning:



For this training algorithm, we produce two separate output sequences at each training iteration: y^s, which is obtained by sampling from the $p(y_t^s|y_1^s, \ldots, y_{t-1}^s, x)$ probability distribution at each decoding time step, and $\hat{y}$, the baseline output, obtained by maximizing the output probability distribution at each time step, essentially performing a greedy search. We define $r(y)$ as the reward function for an output sequence y, comparing it with the ground truth sequence y^* with the evaluation metric of our choice.

$$L_{rl} = (r(\hat{y}) - r(y^s)) \sum_{t=1}^{n'} \log p(y_t^s|y_1^s, \ldots, y_{t-1}^s, x) \qquad (15)$$

Proposed Mixed Training objective function:

Maximum likelihood can assist policy learning to generate more natural summaries. So use combined version:

$$L_{mixed} = \gamma L_{rl} + (1 - \gamma) L_{ml},$$

Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting (2018) (interesting!!)

Model first selects salient sentence and then rewrite them abstractively, so that instead of encoding and attending to every word in the long input sequence, we avoid almost all redundancy issues. We also use actor-critic policy gradient with sentence-level metric rewards to connect these two neural networks and to learn sentence saliency.

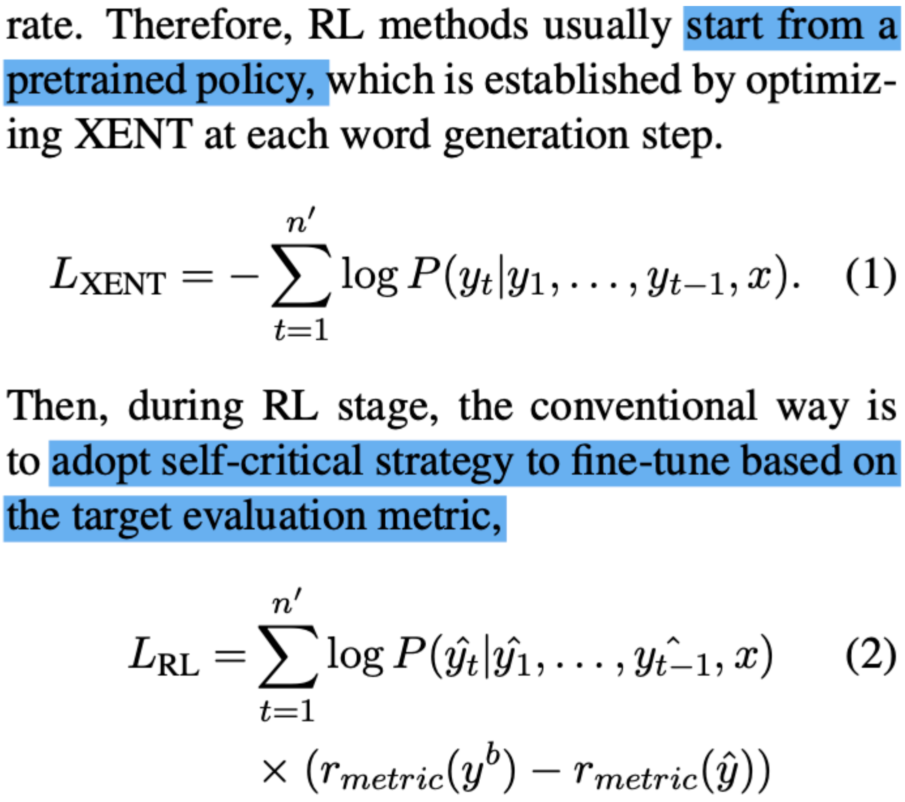
Deep Reinforcement Learning with Distributional Semantic Rewards for Abstractive Summarization (2019)

Introduction:

There exists attention-based architecture, graph techniques and pointer generator in encoder-decoder framework, but long sentence generation has exposure bias as error accumulate during decoding process.

Commonly used automatic evaluation metrics for generating sentence-level rewards is deficient, such as BLEU, ROUGE, suffering from exposure bias, to solve this, many resorts to deep RL with se12seq mode.

However, RL model have low sample efficiency and converge so slow, so can start from a pertained policy, such as optimising XENT, then during RL stage, adopt self-critical strategy to fine-tune based on the target evaluation metric



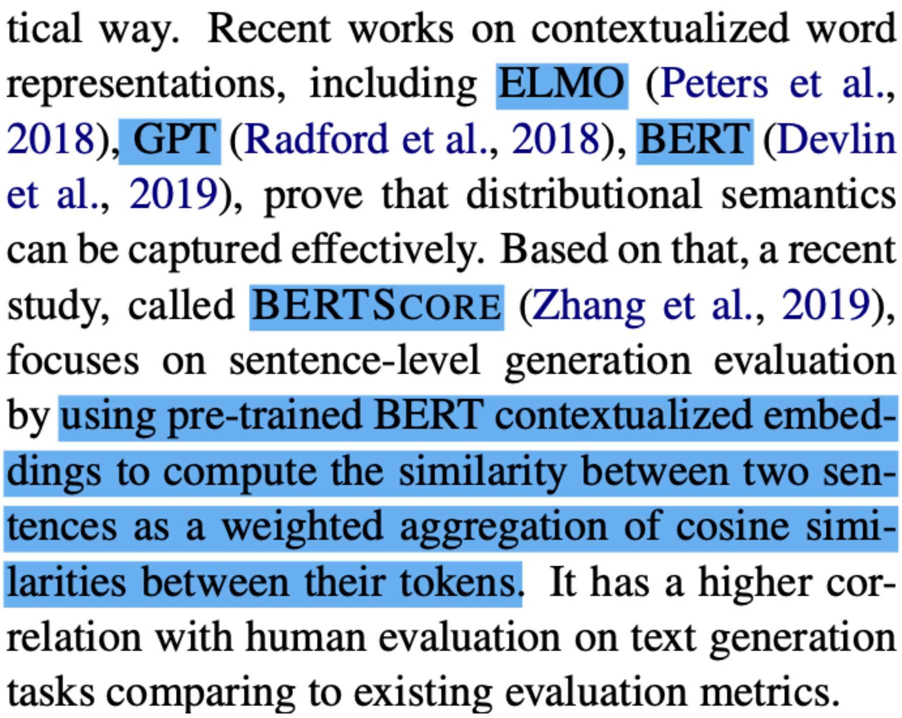
rate. Therefore, RL methods usually start from a pretrained policy, which is established by optimizing XENT at each word generation step.

$$L_{\text{XENT}} = -\sum_{t=1}^{n'} \log P(y_t|y_1, \ldots, y_{t-1}, x). \quad (1)$$

Then, during RL stage, the conventional way is to adopt self-critical strategy to fine-tune based on the target evaluation metric,

$$L_{\text{RL}} = \sum_{t=1}^{n'} \log P(\hat{y_t}|\hat{y_1}, \ldots, y_{\hat{t-1}}, x) \quad (2)$$
$$\times \left(r_{metric}(y^b) - r_{metric}(\hat{y})\right)$$

Distributional Semantic Reward: try to capture the semantic relation between similar words.



tical way. Recent works on contextualized word representations, including ELMO (Peters et al., 2018), GPT (Radford et al., 2018), BERT (Devlin et al., 2019), prove that distributional semantics can be captured effectively. Based on that, a recent study, called BERTSCORE (Zhang et al., 2019), focuses on sentence-level generation evaluation by using pre-trained BERT contextualized embeddings to compute the similarity between two sentences as a weighted aggregation of cosine similarities between their tokens. It has a higher correlation with human evaluation on text generation tasks comparing to existing evaluation metrics.

Proposed Objective Function:

1. DSR + ROUGE
2. DSR + XENT
3. DSR

Deep Transfer Reinforcement Learning for Text Summarization (2019)

Focus mainly on using transfer learning and RL to generalise the model.

The basic underlying summarisation mechanism used is the pointer-generator model, which enable the model to have out-of-vocabulary ability which is necessary for transfer learning.

In RL training, the focus is on minimising the negative expected reward rather than on directly minimising the cross-entropy loss.

$$\mathcal{L}_{Mixed} = (1-\eta)\mathcal{L}_{CE} + \eta\mathcal{L}_{TRL}$$