Adaptive Prior-Dependent Correction Enhanced Reinforcement Learning for Natural Language Generation

Wei Cheng, Ziyan Luo[1], Qiyue Yin[2]

1 Computer Science Department, University of California, San Diego

2 CRISE, Institute of Automation, Chinese Academy of Sciences, Beijing, China

❖ Background & Motivations

- **Neural natural language generation (NLG) aims to generate a piece of new text. NLG models have recently shown remarkable progress in language fluency and coherence. We focus the training of Seq2Seq NLG model.**
  - Neural machine translation (NMT)
  - Image captioning
  - Text summarization
  - …
- **Most of previous works use Maximum Likelihood Estimation (MLE) to train NLG models, but MLE-based training methods suffer from three issues:**
  - Exposure bias: The model is not exposed to the full range of errors during training.
  - Loss inconsistency: During training, we maximize the log-likelihood (token level), but during inference, the model is evaluated by a different metric such as BLEU or ROUGE (sentence level).
  - Deviation ignorance: MLE fails to assign proper scores to different incorrect model outputs, which means that all incorrect outputs are treated equally during training.
- **The main motivation of our research is that Reinforcement Learning (RL) based training method can solve the issues of exposure bias and loss inconsistency, but RL-based training methods also suffer from two issues:**
  - Deviation ignorance: RL-based training methods use the reward such as BLEU or ROUGE also fails to assign proper scores to different incorrect model outputs.
  - Large gradient variance: RL methods such as Policy Gradient have the issue of Large gradient variance, which leads to instability of training.

❖ Methods

- **Advantage weighted Policy Gradient (APG) training:**
  - Advantage function: We introduce the Generalized Advantage Estimation (GAE) method to reduce the variance of gradient.

$$\nabla_\theta L_{RL}(\theta) = \sum_{t=0}^{T} \mathbb{E}_\pi[A^\pi(s_t, a_t)\nabla_\theta log(\pi_\theta(a_t|s_t))] \qquad A^\pi(s_t, a_t) = \sum_{l=1}^{\infty} (\gamma\lambda)^l (r_t + \gamma V^\pi(s_{t+l+1}) - V^\pi(s_{t+l}))$$

  - Estimation of state value: In order to estimate advantage, we need to estimate state value, but there have no dedicated module to estimate state values such as value-network. However, we can estimate state value indirectly.

$$Q^\pi(s_t, a_t) = r_t + \gamma \sum_{s_{t+1} \in S} P(s_{t+1}|s_t, a_t) V^\pi(s_{t+1})$$

  - In NLG task, the discount factor is set to 1 and state transition probability is equal to 1, thus we have

$$A\pi(s_t, a_t) = Q^\pi(s_t, a_t) - Q^\pi(s_{t-1}, a_{t-1})$$

- **Adaptive Prior-Dependent Correction. To alleviate deviation ignorance issue, we enhance the RL objective with a KL term with an adaptive factor:**
  - The prior distribution: We use fast-text to pretrain the word embedding and compute the prior distribution by measuring the cosine distance between tokens and label token:

$$p^*(w_t) = \sigma\left(cos\_sim\left(emb\left(w_t^*\right), emb\left(w_t\right)\right)\right)$$

  - KL loss: KL loss measures the distance between output and prior distribution. We use the Adaptive Factor (negatively correlated to advantage function) to prevent the RL loss from being overcorrected by KL loss:

$$L_{KL}(\theta) = \sum^{T} e^{-\alpha A^\pi(s_t, a_t)} KL\left[p^*(w_t)||p_\theta(w_t)\right] \qquad L(\theta) = -L_{RL}(\theta) + \beta L_{KL}(\theta)$$

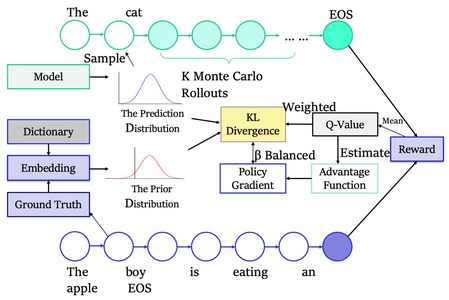

Figure1: Architecture of our approach: APG with APDC

❖ Experiments & Results

- **Our method v.s. baselines**
  - Neural machine translation (NMT)

Method	En-De	En-Zh	Zh-En
Transformer+MLE (Vaswani et al. 2017)	27.30	34.12	24.29
MIXER (Ranzato et al. 2016)	27.43	34.38	24.59
RL4NMT (Wu et al. 2018)	27.52	34.46	24.70
APG	27.63	34.54	24.81
APG+PDC	27.70	34.56	24.90
RL4NMT+PDC	27.81	34.62	24.94
APG+APDC	**28.03**	**34.91**	**25.28**

Table1: BLEU score of NMT for En-De, En-Zh, and Zh-En

  - Image captioning

Method	BLEU-1	BLEU-4	METEOR	ROUGE-L	CIDEr
SCST (Att2all) (Rennie et al. 2017)	-	34.2	26.7	55.7	114.0
OTRL (Chen et al. 2020a)	79.3	34.4	26.8	56.2	111.8
Top-Down+MLE (Anderson et al. 2018)	77.2	36.2	27.0	56.4	113.5
Top-Down+SCST (Anderson et al. 2018)	79.8	36.3	27.7	56.9	120.1
Top-Down+MIXER (Ranzato et al. 2016)	78.4	36.2	27.4	56.5	115.6
Top-Down+SCST+APDC	79.9	36.6	27.8	56.9	120.2
APG	80.1	36.4	28.0	56.9	120.2
APG+PDC	80.3	36.7	28.4	57.3	121.2
APG+APDC (METEOR)	80.1	36.5	29.2	57.1	119.7
APG+APDC	**80.8**	**37.9**	**28.9**	**58.1**	**123.6**

Table2: Performance of image captioning on the MSCOCO Karpathy test split

  - Abstractive text summarization

Method	ROUGE-1	ROUGE-2	ROUGE-L
Pointer (See, Liu, and Manning 2017)	39.53	17.28	36.38
MIXER (Ranzato et al. 2016)	39.78	17.91	37.15
OTRL (Chen et al. 2020a)	41.40	18.22	38.86
APG	41.51	18.34	38.93
APG+PDC	41.60	18.48	39.02
APG+APDC	**42.73**	**18.81**	**39.85**

Table3: Results of abstractive text summarization on CNN/Daily Mail dataset.

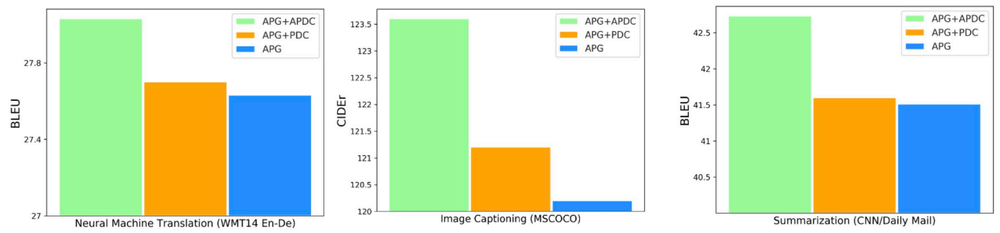

Figure2: Ablation study on three tasks

- Results shows that our method consistently outperforms the existing state-of-art methods on three NLG tasks.
- To evaluate the effectiveness of different components, we compare the results of applying APG, APG+PDC, and APG+APDC on the three tasks.
- The results also show that APG+PDC outperform the APG (other RL), since PDC alleviates the deviation ignorance issue. However, the improvement of PDC is not significant.
- PDC is a token-level objective, but the advantage of RL is that the model can be trained at the sequence level, and PDC weakens this advantage. Therefore, PDC needs a sequence-level adaptive factor to adjust how much the token-level objective affects sequence-level training.
- The results that APG+APDC significantly outperforms other methods also proves the importance of the adaptive mechanism.

❖ Summary

- **We propose a novel technique: adaptive prior-dependent correction (APDC) to further address the deviation ignorance issue that former RL-based approaches on NLG seldom study.**
- **We utilize advantage-function-weighted policy gradient (APG) to work well with APDC, meanwhile alleviate the sparse reward issue. Enhancing APG with APDC can strike a balance between token-level and sequence-level optimization.**
- **Further works: make the algorithm more efficient, …**