Sequence Prediction with Unlabeled Data by Reward Function Learning

[1]Lijun Wu, [2]Li Zhao, [2]Tao Qin, [1]Jianhuang Lai and [2]Tie-Yan Liu

[1] Sun Yat-sen University [2] Microsoft Research Asia

IJCAI-17

1. Motivation

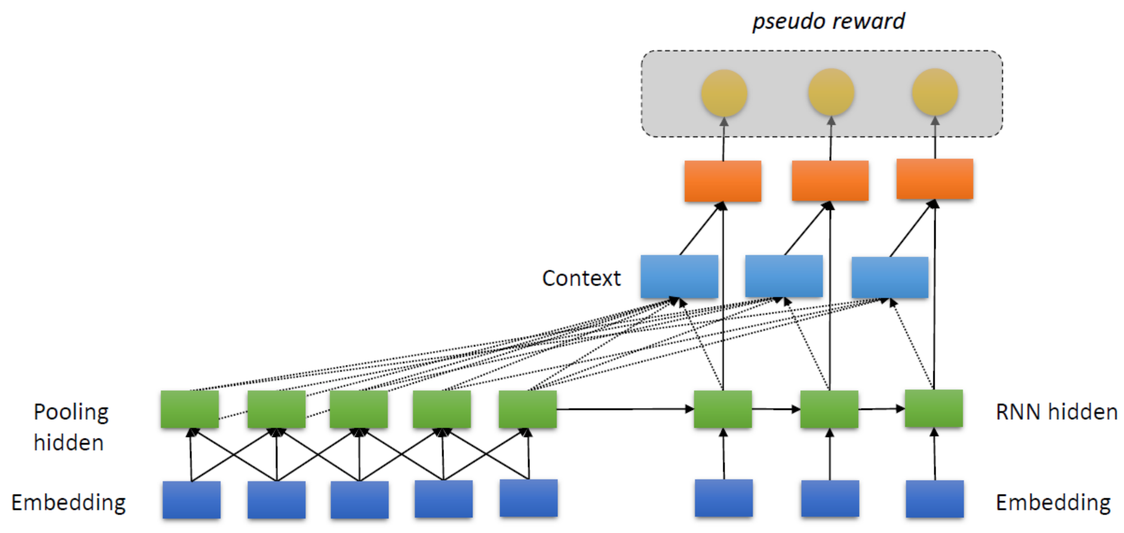
- **Reinforcement Learning** for sequence prediction
  - Introduce reward to optimize the final evaluation metric directly
  - However, reward is defined on ground-truth Y, which limits the approach to labeled data only
- How to extend RL approach to exploit **unlabeled data**?
  - Learn reward function to give pseudo reward

2. Challenge

- Predict pseudo reward based on $(\mathbf{X}, \hat{\mathbf{Y}})$, while true reward is defined on $(Y, \hat{Y})$
- the **sparsity** of non-zero reward for all possible $\hat{Y}$

3. Model

- RNN-based reward network with attention mechanism
  - Take $(X, \hat{Y})$ as input
  - Predict the shaped reward at **every time step**



- MSE training with **biased data distribution**
  - Use sampled $\hat{Y}$ from current policy

$$\mathcal{O}_{MSE}(\theta) = \sum_{i=1}^{N} \sum_{\hat{Y}} \sum_{t=1}^{T} \{g_\theta(\hat{y}_t; \hat{Y}_{1...t-1}, X^{(i)}) - r_t(\hat{y}_t; \hat{Y}_{1...t-1}, Y^{(i)})\}^2$$

Contact

- **wulijun3@mail2.sysu.edu.cn lizo@microsoft.com taoqin@microsoft.com**
- **http://research.microsoft.com/en-us/people/taoqin/**

4. Algorithm

Algorithm 1: REINFORCE Training for Sequence Prediction with Unlabeled Data

Require: An policy $p_\phi(a|\hat{Y}_{1...t}, X)$ and a reward function $g_\theta(a; \hat{Y}_{1...t}, X)$ with weights ϕ and θ respectively; Labeled data set $\{X^{(i)}, Y^{(i)}\}_{i\in\{1,...,N\}}$; Unlabeled data set $\{X^{(i)}\}_{i\in\{N+1,...,N+M\}}$.

Initialize delayed policy p' and delayed reward function g' with same weight: $\phi' = \phi, \theta' = \theta$
while Not Converged **do**
  Receive a random data
  Generate a sequence of actions $\hat{Y}$ from p'
  if the received data is labeled data (X, Y) **then**
    compute shaped reward with ground-truth for all t $r_t(\hat{y}_t; \hat{Y}_{1...t-1}) = R(\hat{Y}_{1...t}, Y) - R(\hat{Y}_{1...t-1}, Y)$
    Update reward function weights using the gradient for all t $\frac{d}{d\theta}(g_\theta(\hat{y}_t; \hat{Y}_{1...t-1}, X) - r_t(\hat{y}_t; \hat{Y}_{1...t-1}, Y))^2$
  else
    compute shaped reward with reward function for all t $r_t(\hat{y}_t; \hat{Y}_{1...t-1}) = \alpha g'(\hat{y}_t; \hat{Y}_{1...t-1}, X)$
  end if
  Compute value function $V_t(\hat{y}_t; \hat{Y}_{1...t-1})$ for all t $V_t(\hat{y}_t; \hat{Y}_{1...t-1}) = \sum_{\tau=t}^{T} r_t(\hat{y}_\tau; \hat{Y}_{1...\tau-1})$
  Update policy weights ϕ using the following gradient estimate $\sum_{t=1}^{T} \frac{d \log p(a=\hat{y}_t|\hat{Y}_{1...t-1}, X)}{d\phi} V_t(\hat{y}_t; \hat{Y}_{1...t-1})$
  Update delayed policy and reward, with a constant γ $\phi' = \gamma\phi + (1-\gamma)\phi', \theta' = \gamma\theta + (1-\gamma)\theta'$
end while

5. Experiments

- **Neural Machine Translation**

Model	Greedy	Beam search
LL∗ [Ranzato *et al.*, 2016]	17.74	20.3
MIXER∗ [Ranzato *et al.*, 2016]	20.73	21.8
LL [Bahdanau *et al.*, 2016]	19.33	21.46
RF [Bahdanau *et al.*, 2016]	20.92	21.35
Semi-supervised baseline	20.10	21.65
Our work	**21.64**	**22.35**

Table 1: BLEU scores for different models on German-English translation test set. LL, RF stands for log-likelihood, REINFORCE. The asterisk identifies results from [Ranzato *et al.*, 2016]. LL and LL∗ both denote maximum log-likelihood training, while LL is implemented by Blocks [Van Merriënboer *et al.*, 2015] and LL∗ is implemented by Torch [Collobert *et al.*, 2011].

- **Text Summarization**

Model	Greedy	Beam search
Log-Likelihood	8.85	10.22
REINFORCE	12.15	12.87
Our work	**12.89**	**13.21**

Table 2: ROUGE-2 scores compared to REINFORCE on text summarization test set.