Reinforcement Learning for Image Captioning

--Pengyu Yan
Back Ground
Image Captioning

Generate natural language description for image

- group of people learning how to surf on the beach
Image Captioning

Previous Work

RNN-based Encoder-Decoder framework
Auto regression way of generating caption
Image Captioning

Motivation

The word that is picked up is only based on the current state

The choice of the best word according to current state may not be the optimal word globally

Lack of global guidance for decision making
Image Captioning

Motivation

The word that is picked up is only based on the current state

The choice of the best word according to current state may not be the optimal word globally

Lack of global guidance for decision making
Image Captioning
Reinforcement Learning Based Method

- Goal: to generate a visual description given an image
- Agent: the image captioning model to learn
- Environment: the given image $I$ + the words predicted so far $\{w_1, ..., w_t\}$
- State: representation of the environment at $t$, $s_t = \{I, w_1, ..., w_t\}$
- Action: the word to generate at $t+1$, $a_t = w_{t+1}$
- Reward: the feedback for reinforcement learning
Proposed Method
We propose a decision-making framework for image captioning.

- An agent model contains a policy network, to capture the local information, and a value network, to capture the global information.
- Training using reinforcement learning with embedding reward.
- Testing using lookahead inference.
Image Captioning

Proposed Method

Policy Net:
- Action based on current state

Value Net:
- Evaluate the policy and serve as global inference guidance
Image Captioning
Implementation

- Policy Network
- Value Network
- Definition of Reward & Reward Network
- Reinforcement Learning -- A2C
Image Captioning
Implementation - Policy Network

ResNet-50 (pre-trained = True)
Linear Layer (1000, 256)
LSTM (1 layer of LSTM cell)
Image Captioning
Implementation - Policy Network

Vocabulary: 9650
9650-class classification

Pretrained with Cross Entropy Loss
Image Captioning

Implementation - Value Net

ResNet-50: Visual Feature Extraction
LSTM: Caption Text feature Extraction

MLP: Process merged feature vector from two domain to generate the **value** according to the policy of current state

{\mathbf{s}_t = \{ \mathbf{I}, w_1, \ldots, w_t \}}

concatenation layer

\[ v_\theta(s_t) \text{ is trying to regress the reward in the end} \]

Pre-Training:
- Regression Problem
- Mean Square Error (MSE) Loss

Regress to **Reward**
Image Captioning
Implementation - Reward Net

**Reward Net:**
Mapping the visual feature and caption text feature into a new embedding space, where:
- **Positive** pair of image and caption would be close to each other in embedding space
- **Negative** pair would be far from each other

Pre-trained with the distance loss of two vector within embedding space

**Reward:**
$R_T = \frac{h_{T-1}(S) \cdot l_m(f_t)}{||h_{T-1}(S) \cdot l_m(f_t)||}$
Image Captioning
Reinforcement - A2C

Pre-trained policy net as Actor
Pre-trained value net as Critic

Well-trained reward network used to calculate reward
(Reward net don’t participate training process here)
Experiment and Result
Inference with Beam Search

\[ W_{t+1} = \underset{w_{b,[t+1]} \in W_{t+1}}{\text{argtopB}} S(w_{b,[t+1]}) \]

\[ S(w_{b,[t+1]}) = S(\{w_{b,[t]}, w_{b,t+1}\}) \]
\[ = S(w_{b,[t]}) + \lambda \log p_\pi(a_t|s_t) + (1 - \lambda) v_\theta(\{s_t, w_{b,t+1}\}) \]
Example of Result

No beam search:
  a bathroom with a toilet sink and mirror

Beam search but without Reinforce:
  a white toilet sitting in front of a white sink in a bathroom

Beam search with Reinforce:
  a white toilet sitting in a bathroom next to a sink and a mirror on the wall
Example of Result

No beam search:
  a city street with cars and cars and motorcycles

Beam search but without Reinforce:
  an image of cars and cars on a city street at night time

Beam search with Reinforce:
  an image of a busy city street with cars parked on the side of it
# Quantitative Evaluation

<table>
<thead>
<tr>
<th>Method</th>
<th>Bleu_1</th>
<th>Bleu_2</th>
<th>Bleu_3</th>
<th>Bleu_4</th>
<th>METEOR</th>
<th>ROUGE_L</th>
<th>CIDEr</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Policy Net Only</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No beam search</td>
<td>0.4865</td>
<td>0.3110</td>
<td>0.1872</td>
<td>0.1123</td>
<td>0.1872</td>
<td>0.3831</td>
<td>0.3494</td>
</tr>
<tr>
<td><strong>Policy Net Only</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>With beam search</td>
<td>0.5010</td>
<td>0.3250</td>
<td>0.1983</td>
<td>0.1197</td>
<td>0.1888</td>
<td>0.3932</td>
<td>0.3637</td>
</tr>
<tr>
<td><strong>A2C Reinforce</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>With beam search</td>
<td>0.5957</td>
<td>0.4040</td>
<td>0.2570</td>
<td>0.1618</td>
<td>0.1841</td>
<td>0.4368</td>
<td>0.5369</td>
</tr>
<tr>
<td>Value as guidance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Quantitative Evaluation Result
Thanks for Listening!