Deep Reinforcement Learning-based Image Captioning with Embedding Reward

Zhou Ren[1], Xiaoyu Wang[1], Ning Zhang[1], Xutao Lv[1], Li-Jia Li[2]
[1]Snap Inc. [2]Google Inc.

IEEE 2017 Conference on Computer Vision and Pattern Recognition
CVPR July 21-26 2017 HONOLULU

Decision-Making Framework for Image Captioning

Limitations of the Encoder-Decoder Framework:

- Only **local** information is utilized in training and inference
- Prone to **accumulate** generation errors during inference
- **Sensitive** to beam sizes in inference

→ *Decision-Making framework with Reinforcement Learning*

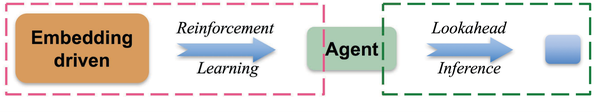
Our Target for the New Framework:

- Better at utilizing **global** information
- **Less likely** to accumulate generation errors during inference
- **Less sensitive** to beam sizes in inference

Problem Re-formulation in the New Framework:

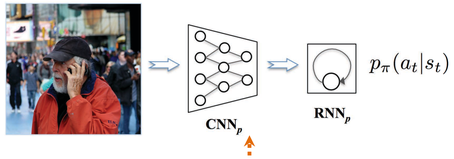
- Agent: the image captioning model to learn
- Environment: an image $\mathbf{I}$, and the words predicted so far
- State s_t: the representation of environment at each time step t
- Action a_t: the word to predict at each time step t

Overview of Our Approach:

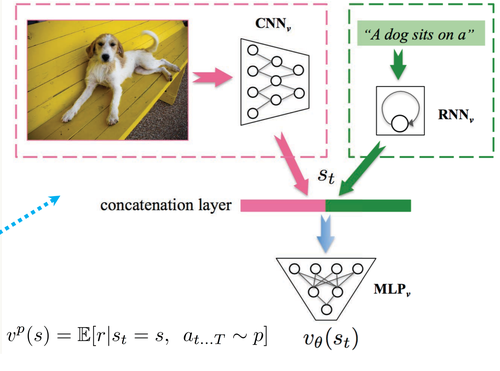


- Agent Design
- Agent Training
- Agent Inference

Agent Design: Policy Network + Value Network

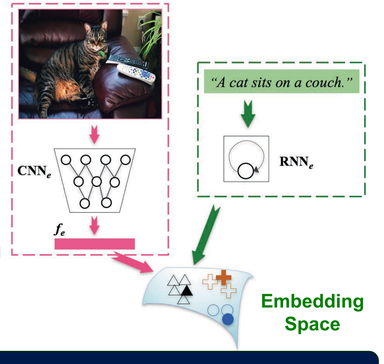


- **Policy Net**: provide the confidence of predicting the next word according to the current state.
  → local guidance
- **Value Net**: evaluate how good a given state is for the final goal.
  → global and lookahead guidance

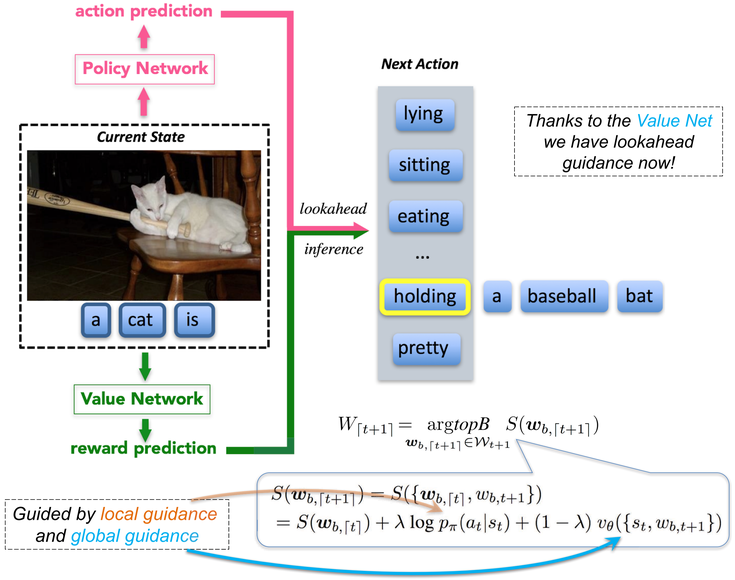


Agent Training: RL with Embedding Reward

- Pretrain Policy Net with cross entropy loss
- Pretrain Value Net with mean squared loss
- Jointly train Policy Net and Value Net using deep Reinforcement Learning
  - an Actor-Critic RL model
  - use MIXER training [Ranzato et al. 2016]
  - reward is defined by visual-semantic embedding



Agent Inference: Lookahead Inference



Experiments on MS-COCO

better at capturing global information | *less likely to accumulate errors*

Quantitative Results and Ablation Study:

Methods	Bleu-1	Bleu-2	Bleu-3	Bleu-4	METEOR	Rouge-L	CIDEr
Google NIC [44]	0.666	0.461	0.329	0.246	–	–	–
m-RNN [30]	0.67	0.49	0.35	0.25	–	–	–
BRNN [17]	0.642	0.451	0.304	0.203	–	–	–
LRCN [7]	0.628	0.442	0.304	0.21	–	–	–
MSR/CMU [3]	–	–	–	0.19	0.204	–	–
Spatial ATT [46]	**0.718**	0.504	0.357	0.25	0.23	–	–
gLSTM [15]	0.67	0.491	0.358	0.264	0.227	–	0.813
MIXER [35]	–	–	–	0.29	–	–	–
Semantic ATT [48] *	0.709	0.537	0.402	**0.304**	0.243	–	–
DCC [13] *	0.644	–	–	–	0.21	–	–
Ours	0.713	**0.539**	**0.403**	**0.304**	**0.251**	**0.525**	**0.937**

Methods	Bleu-1	Bleu-2	Bleu-3	Bleu-4	METEOR	Rouge-L	CIDEr
SL	0.692	0.519	0.384	0.289	0.237	0.512	0.872
SL-Embed	0.7	0.523	0.383	0.280	0.241	0.514	0.888
SL-RawVN	0.706	0.533	0.395	0.298	0.243	0.52	0.916
hid-VN	0.603	0.429	0.292	0.197	0.2	0.467	0.69
hid-Im-VN	0.611	0.435	0.297	0.201	0.202	0.468	0.701
Full-model	0.713	0.539	0.403	0.304	0.251	0.525	0.937

Our approach is modular w.r.t. the agent design

Validated the contribution of all key components in our framework: embedding; reinforcement learning; and lookahead inference.

Parameter Sensitivity Analysis:

λ	Bleu-1	Bleu-2	Bleu-3	Bleu-4	METEOR	Rouge-L	CIDEr
0	0.638	0.471	0.34	0.247	0.233	0.501	0.8
0.1	0.683	0.51	0.373	0.274	0.248	0.516	0.894
0.2	0.701	0.527	0.389	0.288	0.248	0.521	0.922
0.3	0.71	0.535	0.398	0.298	**0.251**	0.524	0.934
0.4	**0.713**	**0.539**	**0.403**	**0.304**	0.247	**0.525**	**0.937**
0.5	0.71	0.538	0.402	0.304	0.246	0.524	0.934
0.6	0.708	0.535	0.399	0.301	0.245	0.522	0.923
0.7	0.704	0.531	0.395	0.297	0.243	0.52	0.912
0.8	0.7	0.526	0.392	0.295	0.241	0.518	0.903
0.9	0.698	0.524	0.389	0.293	0.24	0.516	0.895
1	0.694	0.52	0.385	0.289	0.238	0.513	0.879

Our approach works well due to both the global and local guidance

Method	Beam size	Bleu-1	Bleu-2	Bleu-3	Bleu-4	METEOR	Rouge-L	CIDEr
SL	5	0.696	0.522	0.388	0.29	0.238	0.513	0.876
	10	0.692	0.519	0.384	0.289	0.237	0.512	0.872
	25	0.683	0.508	0.374	0.281	0.234	0.505	0.853
	50	0.680	0.505	0.372	0.279	0.233	0.503	0.850
	100	0.679	0.504	0.372	0.279	0.233	0.503	0.849
Ours	5	0.711	0.538	0.403	0.302	0.251	0.524	0.934
	10	0.713	0.539	0.403	0.304	0.251	0.525	0.937
	25	0.709	0.534	0.398	0.299	0.248	0.522	0.928
	50	0.708	0.533	0.397	0.298	0.247	0.52	0.924
	100	0.707	0.531	0.395	0.297	0.244	0.52	0.92

Our approach is less sensitive to beam sizes comparing to encoder-decoder