International Journal of Research in Engineering, Science and Management
Volume-1, Issue-9, September-2018
www.ijresm.com | ISSN (Online): 2581-5782

Self-Critical Sequence Training for Image Captioning

Pratik Rane[1], Anagha Sargar[2], Faiza Shaikh[3]
[1,2,3]*Student, Department of Computer Engineering, MGMCET, Navi Mumbai, India*

***Abstract*—Image captioning aims at generating a natural language description of an image. Open domain captioning is a very challenging task, as it requires a fine-grained understanding of the global and the local entities in an image, as well as their attributes and relationships. The recently released MSCOCO challenge [1] provides a new, larger scale platform for evaluating image captioning systems, complete with an evaluation server for benchmarking competing methods. Deep learning approaches to sequence modeling have yielded impressive results on the task, dominating the task leaderboard. Inspired by the recently introduced encoder/decoder paradigm for machine translation using recurrent neural networks (RNNs) [2], [3], and [4] use a deep convolutional neural network (CNN) to encode the input image, and a Long Short Term Memory (LSTM) [5] RNN decoder to generate the output caption. These systems are trained end-to-end using back-propagation, and have achieved state-of-the-art results on MSCOCO. More recently in [6], the use of spatial attention mechanisms on CNN layers to incorporate visual context—which implicitly conditions on the text generated so far—was incorporated into the generation process.**

***Index Terms*— Image Captioning, Self-Critical Sequence**

I. Introduction

Deep generative models for text are typically trained to maximize the likelihood of the next ground-truth word given the previous ground-truth word using back-propagation. This approach has been called “Teacher-Forcing” [8]. However, this approach creates a mismatch between training and testing, since at test-time the model uses the previously generated words from the model distribution to predict the next word. This exposure bias [7], results in error accumulation during generation at test time, since the model has never been exposed to its own predictions.

Several approaches to overcoming the exposure bias problem described above have recently been proposed. In [8] they show that feeding back the model’s own predictions and slowly increasing the feedback probability p during training leads to significantly better test-time performance. Another line of work proposes “Professor-Forcing” [9], a technique that uses adversarial training to encourage the dynamics of the recurrent network to be the same when training conditioned on ground truth previous words and when sampling freely from the network.

Recently it has been shown that both the exposure bias and non-differentiable task metric issues can be addressed by incorporating techniques from Reinforcement Learning (RL) [14]. Specifically in [7], Ranzato et al. use the REINFORCE algorithm [15] to directly optimize nondifferentiable, sequence-based test metrics, and overcome both issues. REINFORCE as we will describe, allows one to optimize the gradient of the expected reward by sampling from the model during training, and treating those samples as ground-truth labels (that are re-weighted by the reward they deliver). The major limitation of the approach is that the expected gradient computed using mini-batches under REINFORCE typically exhibit high variance, and without proper context-dependent normalization, is typically unstable.

In this paper we present a new approach to sequence training which we call self-critical sequence training (SCST), and demonstrate that SCST can improve the performance of image captioning systems dramatically. SCST is a REINFORCE algorithm that, rather than estimating the reward signal, or how the reward signal should be normalized, utilizes the output of its own test-time inference algorithm to normalize the rewards it experiences. As a result, only samples from the model that outperform the current test-time system are given positive weight, and inferior samples are suppressed. Using SCST, attempting to estimate the reward signal, as actor-critic methods must do, and estimating normalization, as REINFORCE algorithms must do, is avoided, while at the same time harmonizing the model with respect to its test-time inference procedure. Empirically we find that directly optimizing the CIDEr metric with SCST and greedy decoding at test-time is highly effective. Our results on the MSCOCO evaluation sever establish a new state-of-the-art on the task, improving the best result in terms of CIDEr from 104.9 to 114.7.

II. Captioning Models

In this section we describe the recurrent models that we use for caption generation.

FC models. Similarly to [3] [4], we first encode the input image F using a deep CNN, and then embed it through a linear projection W_I . Words are represented with one hot vectors that are embedded with a linear embedding E that has the same output dimension as W_I. The beginning of each sentence is marked with a special BOS token, and the end with an EOS token. Under the model, words are gen- erated and then fed back into the LSTM, with the image treated as the first word W_I

CN N (*F*). The following up- dates for the hidden units and cells of an LSTM define the model [5]:

$x_t = E1_w t-1$ for $t > 1$, $x_1 = W_I\ CN\ N\ (F\ )$
$i_t = \sigma\ (W_{ix}x_t + W_{ih}h_{t-1} + b_i)$ (Input Gate) f_t
$= \sigma\ (W_{fx}x_t + W_{fh}h_{t-1} + b_f\ )$ (Forget Gate) o_t
$= \sigma\ (W_{ox}x_t + W_{oh}h_{t-1} + b_o)$ (Output Gate)
$ct = it\ Ⓢ\varphi(Wz⊗xxt + Wz⊗hht-1\ + b⊗z\ ) + ft\ Ⓢct-1$
$h_t = o_t\ Ⓢ\tanh(c_t)$
$s_t = W_s h_t$,

where φ is a maxout non-linearity with 2 units (denotes the units) and σ is the sigmoid function. We initialize h_0 and c_0 to zero. The LSTM outputs a distribution over the next word w_t using the softmax function:

$$w_t \sim \text{softmax}\ (s_t) \qquad (1)$$

III. Self-Critical Sequence Training (SCST)

The central idea of the self-critical sequence training (SCST) approach is to baseline the REINFORCE algorithm with the reward obtained by the current model under the inference algorithm used at test time. The gradient of the negative reward of a sample w^s from the model w.r.t. to the softmax activations at time-step t then becomes

$$\frac{\partial L(\theta)}{\partial s_t} = (r(w\ ) - r(w\hat{}))(p_\theta(w_t|h_t) - 1_w s\ ). \quad (9)$$

where $r(w\hat{})$ again is the reward obtained by the current model under the inference algorithm used at test time. Accordingly, samples from the model that return higher reward than $w\hat{}$ will be "pushed up", or increased in probability, while samples which result in lower reward will be suppressed. Like MIXER [7], SCST has all the advantages of REINFORCE algorithms, as it directly optimizes the true, sequence-level, evaluation metric, but avoids the usual scenario of having to learn a (context-dependent) *estimate* of expected future rewards as a baseline. In practice we have found that SCST has much lower variance, and can be more effectively trained on mini-batches of samples using SGD. Since the SCST baseline is based on the test-time estimate under the current model, SCST is forced to improve the performance of the model under the inference algorithm used at test time. This encourages training/test time consistency like the maximum likelihood-based approaches "Data as Demonstrator" [8], "Professor Forcing" [9], and E2E [7], but importantly, it can directly optimize sequence metrics. Finally, SCST is self-critical, and so avoids all the inherent training difficulties associated with actor-critic methods, where a second "critic" network must be trained to estimate value functions, and the actor must be trained on *estimated* value functions rather than actual rewards.

Another important generalization is to utilize the inference algorithm as a critic to replace the learned critic of traditional actor-critic approaches.

We have experimented with both TD-SCST and "True" SCST as described above on the MSCOCO task, but found that they did not lead to significant additional gain. We have also experimented with learning a control-variate for the SCST baseline on MSCOCO to no avail. Nevertheless, we anticipate that these generalizations will be important for other sequence modeling tasks, and policy-gradient-based RL more generally.

IV. Experiments

Dataset. We evaluate our proposed method on the MSCOCO dataset [1]. For offline evaluation purposes we used the data splits from [21]. The training set contains 113, 287 images, along with 5 captions each. We use a set of 5*K* image for validation and report results on a test set of 5*K* images as well, as given in [21]. We report four widely used automatic evaluation metrics, BLEU-4, ROUGEL, METEOR, and CIDEr. We prune the vocabulary and drop any word that has countless then five, we end up with a vocabulary of size 10096 words.

Image Features 1) *FC Models.* We use two type of Features: a) (FC-2k) features, where we encode each image with Resnet-101 (101 layers) [22]. Note that we do not rescale or crop each image. Instead we encode the full image with the final convolutional layer of resnet, and apply average pooling, which results in a vector of dimension 2048. b) (FC-15K) features where we stack average pooled 13 layers of Resnet-101 (11 1024 and 2 2048). These 13 layers are the odd layers of conv4 and conv5, with the exception of the 23rd layer of conv4, which was omitted. This results in a feature vector of dimension 15360.

2) *Spatial CNN features for Attention models:* (Att2in) we encode each image using the residual convolutional neural network Resnet-101 [22]. Note that we do not rescale or crop the image. Instead we encode the full image with the final convolutional layer of Resnet-101, and apply spatially adaptive max-pooling so that the output has a fixed size of 14 × 14 × 2048. At each time step the attention model produces an attention mask over the 196 spatial locations. This mask is applied and then the result is spatially averaged to produce a 2048 dimension representation of the attended portion of the image.

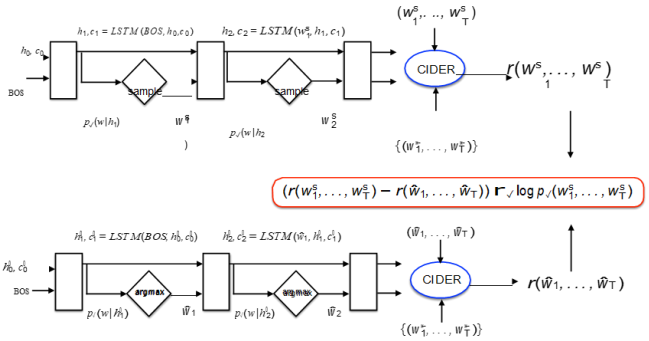

Fig. 1. Self-critical sequence training (SCST)

The weight put on words of a sampled sentence from the

model is determined by the difference between the reward for the sampled sentence and the reward obtained by the estimated sentence under the test-time inference procedure (greedy inference depicted). This harmonizes learning with the inference procedure, and lowers the variance of the gradients, improving the training procedure.

Implementation Details. The LSTM hidden, image, word and attention embeddings dimension are fixed to 512 for all of the models discussed herein. All of our models are trained according to the following recipe, except where otherwise noted. We initialize all models by training the model under the XE objective using ADAM [20] optimizer with an initial learning rate of $5\times10-4$. We anneal the learning rate by a factor of 0.8 every three epochs, and increase the probability of feeding back a sample of the word posterior by 0.05 every 5 epochs until we reach a feedback probability 0.25 [8]. We evaluate at each epoch the model on the development set and select the model with best CIDEr score as an initialization for SCST training. We then run SCST training initialized with the XE model to optimize the CIDEr metric (specifically, the CIDEr-D metric) using ADAM with a learning rate $5 \times 10-5$. Initially when experimenting with FC-2k and FC-15k models we utilized curriculum learning (CL) during training, as proposed in [7], by increasing the number of words that are sampled and trained under CIDEr by one each epoch (the prefix of the sentence remains under the XE criterion until eventually being subsumed). We have since realized that for the MSCOCO task CL is not required, and provides little to no boost in performance. The results reported here for the FC-2K and FC-15K models are trained with CL, while the attention models were trained directly on the entire sentence for all epochs after being initialized by the XE seed models.

TABLE I
PERFORMANCE OF SELF-CRITICAL SEQUENCE TRAINING

Training Metric	Evaluation Metric CIDEr	BLEU4	ROUGEL	METEOR
XE	90.9	28.6	52.3	24.1
XE (beam)	94.0	29.6	52.6	25.2
MIXER-B	101.9	30.9	53.8	24.9
MIXER	104.9	31.7	54.3	25.4
SCST	106.3	31.9	54.3	25.5

The Table-1, Performance of self-critical sequence training (SCST) versus MIXER [7] and MIXER without a baseline (MIXER-B) on the test portion of the Karpathy splits when trained to optimize the CIDEr metric (FC-2K models). All improve the seed cross-entropy trained model, but SCST outperforms MIXER.

V. EXAMPLE OF GENERATED CAPTIONS

Here we provide a qualitative example of the captions generated by our systems for the image in figure6. This picture is taken from the objects out-of-context (OOOC) dataset of images [24]. It depicts a boat situated in a un- usual context, and tests the ability of our models to compose descriptions of images that differ from those seen during training. The top 5 captions returned by the XE and SCST trained FC-2K, FC-15K, and attention model ensembles when deployed with a decoding "beam" of 5 are depicted in figure7 3. On this image the FC models fail completely, and the SCST-trained ensemble of attention models is the only system that is able to correctly describe the image. In general, we found that the performance of all captioning systems on MSCOCO data is qualitatively similar, while on images containing objects situated in an uncommon context [24] (i.e. unlike the MSCOCO training set) our attention models perform much better, and SCST-trained attention models output yet more accurate and descriptive captions. In general, we qualitatively found that SCST-trained attention models describe images more accurately, and with higher confidence, as reflected in Figure7, where the average of the log-likelihoods of the words in each generated caption are also depicted. Additional examples can be found in the supplementary material. Note that we found that Att2in attention models actually performed better than our Att2all models when applied to images "from the wild", so here we focus on demonstrating them.

TABLE II
SINGLE BEST MODELS (XE)

Model Type	Search Method	Evaluation Metric CIDEr	BLEU4	ROUGEL	METEOR
FC-2K	greedy	90.9	28.6	52.3	24.1
	beam	94.0	29.6	52.6	25.2
FC-15K	greedy	94.1	29.5	52.9	24.4
	beam	96.1	30.0	52.9	25.2
Att2in	greedy	99.0	30.6	53.8	25.2
	beam	101.3	31.3	54.3	26.0
Att2all	greedy	97.9	29.3	53.4	25.4
(RL seed)	beam	99.4	30.0	53.4	25.9

TABLE III
SINGLE BEST MODELS (SCST UNLESS NOTED O.W.)

Model Type	Search Method	Evaluation Metric CIDEr	BLEU4	ROUGEL	METEOR
FC-2K	greedy	106.3	31.9	54.3	25.5
	beam	106.3	31.9	54.3	25.5
FC-15K	greedy	106.4	32.2	54.6	25.5
	beam	106.6	32.4	54.7	25.6
Att2in	greedy	111.3	33.3	55.2	26.3
	beam	111.4	33.3	55.3	26.3
4 Att2all (REINF.)	greedy	110.2	32.7	55.1	26.0
	beam	110.5	32.8	55.2	26.1
4 Att2all (MIXER-CL)	greedy	112.9	34.0	55.5	26.4
	beam	113.0	34.1	55.5	26.5
Att2all	greedy	113.7	34.1	55.6	26.6
	beam	114.0	34.2	55.7	26.7

Table 2: Performance of the best XE and corr. SCST-trained

single models on the Karpathy test split (best of 4 random seeds). The results obtained via the greedy decoding and optimized beam search are depicted. Models learned using SCST were trained to directly optimize the CIDEr metric.

TABLE IV
ENSEMBLED MODELS (XE)

Model Type	Search Method	Evaluation Metric CIDEr	BLEU4	ROUGEL	METEOR
4 FC-2K	greedy	96.3	30.1	53.5	24.8
	beam	99.2	31.2	53.9	25.8
4 FC-15K	greedy	97.7	30.7	53.8	25.0
	beam	100.7	31.7	54.2	26.0
4 Att2in	greedy	102.8	32.0	54.7	25.7
	beam	106.5	32.8	55.1	26.7
Att2all (RL seeds)	greedy	102.0	31.2	54.4	26.0
	beam	104.7	32.2	54.8	26.7

MIXER less CL results (MIXER-) are also included.

TABLE V
ENSEMBLED MODELS (SCST UNLESS O.W. NOTED)

Model Type	Search Method	Evaluation Metric CIDEr	BLEU4	ROUGEL	METEOR
4 FC-2K	greedy	108.9	33.1	54.9	25.7
	beam	108.9	33.2	54.9	25.7
4 FC-15K	greedy	110.4	33.4	55.4	26.1
	beam	110.4	33.4	55.4	26.2
4 Att2in	greedy	114.7	34.6	56.2	26.8
	beam	115.2	34.8	56.3	26.9
4 Att2all (REINF.)	greedy	113.8	34.2	56.0	26.5
	beam	113.6	33.9	55.9	26.5
4 Att2all (MIXER-CL)	greedy	116.6	34.9	56.3	26.9
	beam	116.7	35.1	56.4	26.9
4 Att2all	greedy	116.8	35.2	56.5	27.0
	beam	117.5	35.4	56.6	27.1

TABLE VI
ENSEMBLED MODELS (SCST UNLESS O.W. NOTED)

Ensemble SCST models	Evaluation Metric CIDEr	BLEU4	ROUGEL	METEOR
Ens. 4 (Att2all)	114.7	35.2	56.3	27.0
Ens. 4 (Att2in)	112.3	34.4	55.9	26.8
Previous best	104.9	34.3	55.2	26.6

Table 3: Performance of Ensembled XE and SCST-trained models on the Karpathy test split (ensembled over 4 random seeds). The models learned using self-critical sequence training (SCST) were trained to optimize the CIDEr met- ric. MIXER less CL results (MIXER-) are also included.

Table 4: Performance of 4 ensembled attention models trained with self-critical sequence training (SCST) on the official MSCOCO evaluation server (5 reference captions). The previous best result on the leader board (as of 04/10/2017) is also depicted (http://mscoco.org/dataset/#captions-leaderboard, Table C5, Watson Multimodal).

Fig. 2. Picture of a common object in MSCOCO (a giraffe) situated in an uncommon context (Out of COCO domain) [24]

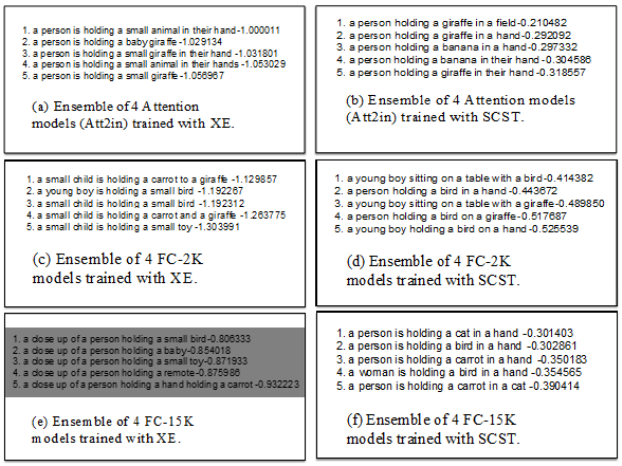

Fig. 3. Captions generated by various models

VI. DISCUSSION AND FUTURE WORK

In this paper we have presented a simple and efficient approach to more effectively base lining the REINFORCE algorithm for policy-gradient based RL, which allows us to more effectively train on non-differentiable metrics, and leads to significant improvements in captioning performance on MSCOCO—our results on the MSCOCO evaluation sever establish a new state-of-the-art on the task. The self-critical approach, which normalizes the reward obtained by sampled sentences with the reward obtained by the model under the test-time inference algorithm is intuitive, and avoids having to estimate both action-dependent and action-independent reward functions.

VII. CONCLUSION

This work has focused on optimizing the CIDEr metric, since, as discussed in the paper, optimizing CIDER substantially improves all MSCOCO evaluation metrics, as was shown in tables 4 and 5 and is depicted in figure 8. Nevertheless, directly optimizing another metric does lead to

higher evaluation scores on that same metric as shown, and so we have started to experiment with including models trained on Bleu, Rouge-L, and METEOR in our Att2in ensemble to attempt to improve it further. So far, we have not been able to substantially improve performance w.r.t. the other metrics without more substantially degrading CIDEr.

REFERENCES

[1] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dolla´r, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. EECV, 2014.1,4

[2] Kyunghyun Cho, Bart van Merrienboer, C¸ aglar G u¨lc¸ehre, Fethi Bougares, Holger Schwenk, and Yoshua Ben- gio. Learning phrase representations using RNN encoder- decoder for statistical machine translation. EMNLP, 2014. 1

[3] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Du- mitru Erhan. Show and tell: A neural image caption gen- erator. CVPR, 2015.1,2

[4] Andrej Karpathy and Fei-Fei Li. Deep visual-semantic align- ments for generating image descriptions. CVPR, 2015.1,2

[5] Sepp Hochreiter and J u¨rgen Schmidhuber. Long short-term memory. Neural Comput., 1997.1,2

[6] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In ICML, 2015.1,2,3

[7] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. ICLR, 2015.1,2,3,4,5,6,11

[8] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In NIPS, 2015.1,4,5,11

[9] Alex Lamb, Anirudh Goyal, Ying Zhang, Saizheng Zhang, Aaron Courville, and Yoshua Bengio. Professor forcing: A new algorithm for training recurrent networks. Neural Infor- mation Processing Systems (NIPS) 2016, 2016.1,4

[10] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In ACL, 2002.2

[11] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In Text summarization branches out: Proceedings of the ACL-04 workshop, volume 8. Barcelona, Spain, 2004.2

[12] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with hu- man judgments. In Proceedings of the acl workshop on in- trinsic and extrinsic evaluation measures for machine trans- lation and/or summarization, volume 29, pages 65–72, 2005.

[13] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evalua- tion. In CVPR, 2015.2,11

[14] Richard S. Sutton and Andrew G. Barto.Reinforcement learning: An introduction. MIT Press, 1998.2,3

[15] Ronald J. Williams. Simple statistical gradient-following al- gorithms for connectionist reinforcement learning. In Machine Learning, pages 229–256, 1992.

[16] John Schulman, Philipp Moritz, Sergey Levine, Michael Jor- dan, and Pieter Abbeel. High-dimensional continuous con- trol using generalized advantage estimation. arXiv preprint arXiv:1506.02438, 2015.2

[17] Wojciech Zaremba and Ilya Sutskever. Reinforcement learn- ing neural turing machines. Arxiv, 2015.2,3

[18] Andriy Mnih and Karol Gregor. Neural variational inference and learning in belief networks. arXiv preprint arXiv:1402.0030, 2014.2

[19] Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron C. Courville, and Yoshua Bengio. An actor-critic algorithm for sequence pre- diction. Arxiv, 2016.2

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. ICLR, 2015.3,5

[21] Andrej Karpathy and Fei-Fei Li. Deep visual-semantic align- ments for generating image descriptions. In CVPR, 2015.4, 6,7

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, 2016.4

[23] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Du- mitru Erhan. Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge. PAMI, 2016.7

[24] Myung Jinchoi, Antonio Torralba, and Alan S. Willsky. Con- text models and out-of-context objects.7,9,11,13,15,16