A DUAL PREDICTION NETWORK FOR IMAGE CAPTIONING

Yanming Guo[1], *Yu Liu*[2], *Maaike H.T. de Boer*[3], *Li Liu*[1,4], *Michael S. Lew*[2]

[1] College of System Engineering, National University of Defense Technology, Changsha, China
[2] LIACS Media Lab, Leiden University, Leiden, the Netherlands
[3] TNO, Anna van Buerenplein 1, the Hague, the Netherlands
[4] CMVS, University of Oulu, Oulu, Finland

ABSTRACT

General captioning practice involves a single forward prediction, with the aim of predicting the word in the next timestep given the word in the current timestep. In this paper, we present a novel captioning framework, namely Dual Prediction Network (DPN), which is end-to-end trainable and addresses the captioning problem with dual predictions. Specifically, the dual predictions consist of a forward prediction to generate the next word from the current input word, as well as a backward prediction to reconstruct the input word using the predicted word. DPN has two appealing properties: 1) By introducing an extra supervision signal on the prediction, DPN can better capture the interplay between the input and the target; 2) Utilizing the reconstructed input, DPN can make another new prediction. During the test phase, we average both predictions to formulate the final target sentence. Experimental results on the MS COCO dataset demonstrate that, benefiting from the reconstruction step, both generated predictions in DPN outperform the predictions of methods based on the general captioning practice (single forward prediction), and averaging them can bring a further accuracy boost. Overall, DPN achieves competitive results with state-of-the-art approaches, across multiple evaluation metrics.

Index Terms— Dual Prediction Network, reconstruction, deep supervision, image captioning

1. INTRODUCTION

Image captioning is an emerging challenge in Vision-to-Language (V2L) research and has been studied extensively in recent years. The purpose of this task is to translate visual images into sensible sentences, which not only reflect objects appearing in images, but also describe their spatial configurations, attributes, as well as the activities.

The early research in image captioning tends to utilize the retrieval-based approaches [1, 2] which first retrieve the closest matching images, and then transfer their captions to the query image. Without learning a language model, these approaches can return decent descriptions for the query images, but they are unable to generate new captions for unseen scenes and are highly dependent on the training data. Recently, more and more attention has been paid to the generative-based approaches [3–8], which can create novel captions rather than retrieving existing ones. The typical pipeline for these approaches consists of a Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN). The CNN is used to extract image features, and the RNN, typically implemented with long short-term memory (LSTM) unit [9], seeks to synthesize semantically meaningful captions. Currently, there is extensive research focusing on how to generate a more appropriate image feature to predict the image caption. An intuitive scheme for improving the image feature is to employ more advanced CNN models, such as VGGNet [10] and ResNet [11]. In addition to compressing the image into a static representation for the RNN, numerous works [5, 8, 12–14] suggest to dynamically adapt the salient features to the forefront in different time steps and have proposed diverse attention models.

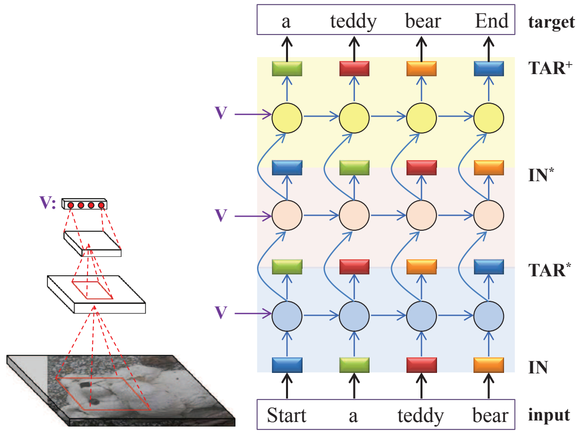


Fig. 1: The overview of the proposed DPN framework. The left side is the CNN part to extract the visual feature V, which is then fed to each LSTM. IN: the embedded encoding of the input sentence. IN*: the reconstructed encoding of the input sentence. TAR*: the first prediction of DPN. TAR$^+$: the second prediction of DPN.

The aforementioned methods handle the image captioning task through a unidirectional prediction, but ignore its reverse prediction, which may optimize the generated descriptions

978-1-5386-1737-3/18/$31.00 ©2018 IEEE

from the other side. The bidirectional captioning idea has ever been investigated in [7], which proposed Bi-directional LSTM (Bi-LSTM) to make use of history and future context information for image captioning. However, their Bi-LSTM is implemented with two independent LSTMs, and it actually generates two independent sentences of different orders. For each individual sentence, it is still a unidirectional prediction.

In this work, we propose a Dual Prediction Network (DPN) to jointly optimize the forward prediction and the backward prediction, as shown in Fig. 1. There are three components in DPN: the bottom blue region IN $\rightarrow$ TAR* is the **prediction** part, which denotes the general captioning practice and formulates the caption through a unidirectional prediction. The middle pink region TAR* $\rightarrow$ IN* is the **reconstruction** part, which strives to reconstruct the input sentence through the prediction. The top yellow region IN* $\rightarrow$ TAR$^+$ is the **re-prediction** part, which aims to get another high-quality prediction using the reconstructed input.

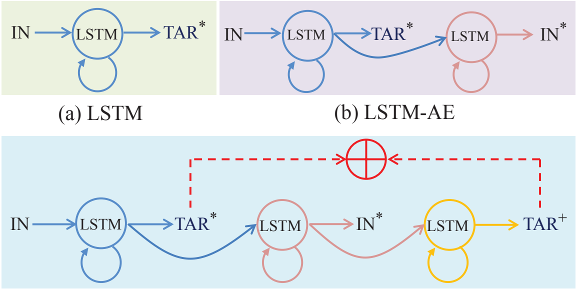
The benefits of DPN are twofold: on the one hand, it introduces an additional supervision on TAR*, enabling the TAR* to be optimized from two directions, i.e. IN $\rightarrow$ TAR* and TAR* $\rightarrow$ IN*, thus guarantees TAR* can achieve better performance. From another perspective, by constraining IN* has a small reconstruction error, it can be employed as a transformation of the input sentence, and generate an additional prediction, i.e. TAR$^+$. During the test phase, we can combine both predictions to boost the performance.

2. RELATED WORK

Image captioning has received a surge of research interest in recent years. We are particularly interested in the CNN-RNN-based literatures for caption generation, as they are most relevant to our work. Such models typically extract the image feature using a CNN, and then send the feature to a RNN for caption generation.

The differences of various models mainly lie in how to employ the image feature. For example, NIC [3] proposed to utilize the image feature once at the first time step for a better initialization, while LRCN [4] employed the image feature at each time step. Aside from feeding a static image feature into the RNN, various attention models have been proposed to dynamically adjust the image feature during the caption generation process. For instance, Xu et al. [5] proposed two attention-based mechanisms, i.e. soft attention & hard attention, to learn where to focus in the image at each time step. This work was extended by [14] to improve the correctness of the visual attention. In addition to the spatial attention, one more recent work, SCA-CNN [8], incorporated channel-wise attention in a CNN and achieved promising performance.

As the image features of the CNN part have been extensively exploited for the image captioning task, there is quite limited work on how to generate better captions of the RNN part. In this work, we propose to supervise the prediction from two directions, and jointly optimize the forward as well as the backward prediction. Some works have ever investigated the bi-directional mapping from one domain to another. For instance, Chen et al. [15] proposed a bi-directional model that is capable of generating captions from visual features, as well as reconstructing visual features given a description. Rohrbach et al. [16] intended to localize textual phrases in visual content, and utilized an attention mechanism to reconstruct the given phrases. Feng et al. [17] constructed a dual space to compare the image and text features, and minimized the reconstruction error from the dual space to the original space. The main purpose of these approaches is to learn a good association across the multiple modalities. In contrast, our proposed framework aims to associate the input/target sentences, therefore, we only need to fulfill the reconstruction within one modality, i.e. text. This formulation enables efficient and intuitive inference without resorting to other complex projections or assumptions.



(a) LSTM (b) LSTM-AE

(c) DPN

Fig. 2: The comparison of LSTM, LSTM-AE and DPN. LSTM denotes the general captioning practice which generates the prediction through a unidirectional prediction. LSTM-AE introduces the backward prediction to reconstruct the input from the target TAR*, and the whole procedure performs like an Auto-Encoder. DPN makes use of the transformation IN* to generate another prediction TAR$^+$, and averages both predictions to generate the final sentence during the test phase, as denoted by the dash red line.

3. DUAL PREDICTION NETWORK

3.1. Overview

The overall schematic framework of DPN is shown in Fig. 2 (c). The framework consists of three LSTM modules, in which the first LSTM takes the embedded input sentence as input, and the remaining two LSTMs take the output of the previous LSTM as input.

The first and the third LSTM aim to make the sentence prediction, therefore, both of them employ the target sentence as the supervision signal. During the test phase, we combine both predictions to formulate the final prediction of DPN, as shown by the red dashed line in Fig. 2 (c). The second LSTM intends to make the reconstruction, so it utilizes the input sentence as the supervision signal.

3.2. Model Formulation

Given an image and its corresponding caption $S = \{s_1, ..., s_T\}$, the input sentence can be written as $S_I = \{s_0, s_1, ..., s_T\}$, and the target sentence is $S_P = \{s_1, ..., s_T, s_{T+1}\}$. Each word s_t is a 1-of-D ('one-hot') encoding vector, with D the size of the vocabulary. The s_0 in S_I and the s_{T+1} in S_P denote the 'Start' and 'End' of the sentence, and they are normally implemented with zero index.

We first utilize CNN to extract the image feature V. As suggested in LRCN [4, 18], the visual feature would be propagated into each time step for the LSTM modules. Next, we linearly embed each word s_t into an n-dimensional real-valued vector $x_t = E_m s_t$, where E_m is a word embedding matrix (learned). Finally, we get an embedded input sentence $X = \{x_0, x_1, ..., x_T\}$, with each element x_t is a column of E_m chosen by the one-hot s_t. For each LSTM module, we aim to maximize the probability of the corresponding target description, thus define the objective of DPN as:

$$\Theta^\star = \underset{\Theta}{\text{argmax}}(\sum \log p^*(S_P|V, X, \Theta) + \sum \log p^r(S_I|V, LSTM_1, \Theta) + \sum \log p^+(S_P|V, LSTM_2, \Theta)) \tag{1}$$

Where Θ is the parameter of the model. p^* and p^+ are the probabilities of the target sentence, and p^r is the probability of the input sentence. $LSTM_1$ and $LSTM_2$ are the outputs of the first and the second LSTM modules.

Using the chain rule, the log likelihood of each probability distribution can be decomposed into ordered conditionals:

$$\log p(S) = \sum_{t=1}^{T} \log p(S_t|S_0, ..., S_{t-1}) \tag{2}$$

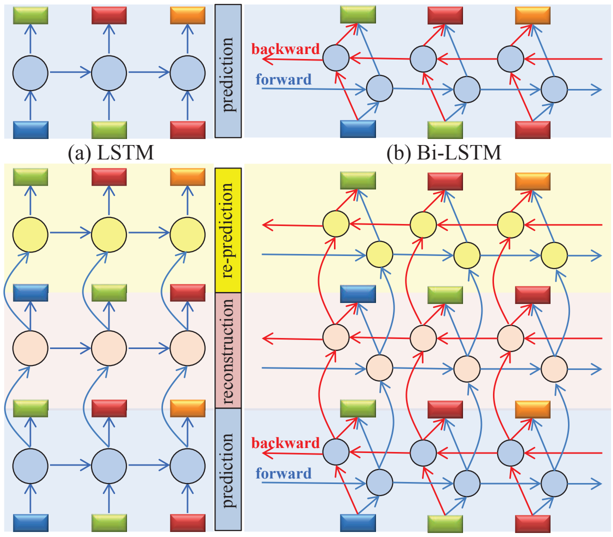
Each conditional is specified as: $p(S_t|S_0, ..., S_{t-1}) \sim softmax(h_t W)$, where W is the weight matrix connecting the hidden state h_t with the distribution over words, and h_t is recursively updated through $h_t = LSTM(x_{t-1}, h_{t-1}, V)$. For the details of the LSTM function, we refer the reader to several recent image captioning works [3, 4].

To maximize the joint probabilities in Eq. 1, we define the loss function as the sum of negative log likelihood of the correct word at each time step:

$$L = -\sum_{t=1}^{T} (\log p_t^*(s_t) + \log p_t^r(s_{t-1}) + \log p_t^+(s_t)) \tag{3}$$

The first LSTM and the third LSTM aim to maximize the probability of the target sentence, while the second LSTM aims to maximize the probability of the input sentence. As the input sentence S_I is one time step prior to the target sentence S_P, for time step t, we choose the probability of word s_{t-1} for the second term.

During the training phase, the loss function is minimized through adjusting the parameters of the model, i.e. Θ. During the test phase, DPN would generate two predictions (i.e. $softmax(h_t W)$), in which we can choose either one as our final prediction, and we can also average their predictions for a further performance boost.



(a) LSTM (b) Bi-LSTM

(c) DPN (d) Bi-DPN

Fig. 3: The demonstration of DPN with LSTM and Bi-LSTM. Both DPN and Bi-DPN consists of three components: prediction, reconstruction and re-prediction.

3.3. Comparison with other models

Comparison with LSTM: The generic CNN-LSTM captioning practice, demonstrated by Fig. 2 (a), is the first part of the proposed DPN, in which the 'one-hot' input sentence is embedded into a 'real-valued' vector through the word embedding matrix E_m, and then propagated into a LSTM for caption generation. Although E_m can be automatically learned by optimizing the network, we cannot ensure the prediction is good enough with a single forward pass of the LSTM.

Comparison with LSTM-AE: LSTM-AE is the intermediate transition model between LSTM and DPN, as is shown in Fig. 2 (b). Compared to the generic CNN-LSTM framework, it additionally introduces a supervision to the prediction process, by explicitly requiring the prediction can reconstruct the input well. This enables the prediction can be optimized from two directions. In addition, the reconstructed input coding of LSTM-AE can be employed for another prediction, as is done by our proposed DPN.

Comparison with Bi-LSTM: Bi-LSTM generates the forward and backward predictions separately through two independent LSTMs, and chooses the one with higher probability as the final prediction. In contrast, DPN jointly optimizes the forward and backward predictions through two cascading LSTMs. Bi-LSTM and DPN are not mutually exclusive, and they can be combined for a stronger bi-directional mapping. In Fig. 3, we demonstrate how we can adapt our DPN model on top of LSTM and Bi-LSTM. For Bi-LSTM, we employ

DPN on each individual prediction process, and get two corresponding predictions. The final prediction is the one with higher probability.

4. EXPERIMENTS

4.1. Experimental setup and implementation details

We conduct our experiments on the international benchmark MS COCO dataset [19], which contains 82783 training images, 40504 validation images and 40775 testing images. Each image has at least five human-annotated captions. Following the normal benchmarking procedure [7, 8, 20], we use the whole training set for training, and choose 5000 images from the validation set for testing.

Visual feature. In order to make a fair comparison, we follow most relevant works and employ the 16-layer VGGNet [10] in the CNN part. We extract the 1000-Dim activations from the last fully-connected layer as the visual feature.

Textual feature. We first build the word vocabulary by performing basic tokenization and removing the words that appear less than 5 times in the training set. The vocabulary contains 8800 words. Next, we represent each word in the sentence as a one-hot vector, and utilize an embedding matrix E_m to encode the vector as the textual feature.

Evaluation Metrics. We use BLEU (B-1,B-2,B-3,B-4) [21], METEOR (M) [22], ROUGE-L (R) [23] and CIDEr (C) [24] as evaluation metrics. All metrics are computed with the public available MS COCO evaluation code.

Training Details. We first utilize the off-the-shelf CNN feature to train the RNN part for 5×10^4 iterations, in order to examine our proposed scheme. Next, we jointly fine-tune the CNN and RNN for another 5×10^4 iterations to achieve better performance. Other configurations are the same with the work in LRCN [4]. All of the experiments are conducted within the Caffe framework.

4.2. Experimental results

Evaluation of the DPN model with LSTM. In this part, we incorporate the DPN module with the unidirectional LSTM and test its performance, as shown in Table 1. We can observe: (1) LSTM-AE and DPN-1 obtained similar performance, suggesting whether we utilize the reconstructed input coding to predict or not, it does not affect the original prediction greatly; (2) Both DPN-1 and DPN-2 achieved better performance than solely utilizing LSTM, demonstrating that it is beneficial to constrain the prediction to reconstruct the input, and the reconstructed input is also more advantageous in predicting the target; (3) The DPN gained remarkable improvement over LSTM. For example, for all of the BLEU metrics, DPN improved the accuracy by about 2 percent over LSTM. This verifies the effectiveness of our proposed model.

In Fig. 4, we present an image caption example from the MS COCO dataset. As can be seen, DPN can effectively incorporate the DPN-1 and DPN-2 predictions, and generates closer result with the human-written groundtruth caption.

Table 1: The performance of LSTM, LSTM-AE and DPN. All of the models are trained with off-the-shelf fc8 activation of VGGNet. DPN-1: the first prediction of DPN; DPN-2: the second prediction of DPN; DPN: the prediction by averaging DPN-1 and DPN-2. Here we use a beam search of size 1.

	B-1	B-2	B-3	B-4	M	R	C
LSTM	64.9	47.1	33.1	22.9	21.3	48.2	70.1
LSTM-AE	65.7	48.0	33.9	23.5	21.5	48.3	70.9
DPN-1	65.6	48.0	33.9	23.6	21.4	48.3	71.0
DPN-2	66.3	48.4	34.1	23.6	21.3	48.3	70.7
DPN	**67.1**	**49.5**	**35.2**	**24.5**	**21.7**	**48.9**	**73.1**

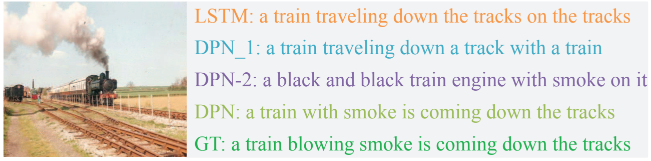

LSTM: a train traveling down the tracks on the tracks
DPN_1: a train traveling down a track with a train
DPN-2: a black and black train engine with smoke on it
DPN: a train with smoke is coming down the tracks
GT: a train blowing smoke is coming down the tracks

Fig. 4: Example of image captioning using different schemes. GT means the groundtruth caption of the image.

Evaluation of the DPN model with Bi-LSTM. In this part, we evaluate the performance DPN model on Bi-LSTM, and report the results in Table 2, from which we can notice that: (1) For the BLEU metric, the DPN module improved the accuracy by about 1 percent for forward/backward/bi-directional predictions. This demonstrates that, our proposed DPN can also be effectively incorporated with the Bi-LSTM; (2) Although the Bi-DPN achieved better performance than Bi-LSTM over all evaluation metrics, the advantage is not as large as LSTM. The reason is that Bi-LSTM shares similar motivation with DPN, incorporating them may be a little redundant. Nevertheless, it should not decrease the awareness of the effectiveness of our proposed DPN.

In Fig. 5, we present an example image to show the generated forward/backward captions of Bi-LSTM and Bi-DPN, in which Bi-DPN can generate a more sensible caption.

Evaluation of end-to-end fine-tuning. We now consider the effect of jointly fine-tuning CNN and RNN components. As can be seen in Table 3, both DPN and Bi-DPN are signif-

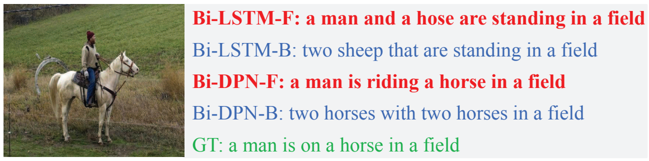

Bi-LSTM-F: a man and a hose are standing in a field
Bi-LSTM-B: two sheep that are standing in a field
Bi-DPN-F: a man is riding a horse in a field
Bi-DPN-B: two horses with two horses in a field
GT: a man is on a horse in a field

Fig. 5: Example of image captioning using Bi-LSTM and Bi-DPN. In both models, forward caption (in red color) is selected as final caption for corresponding image. GT means the groundtruth caption of the image.

Table 2: The performance of Bi-LSTM and Bi-DPN. Both models are trained with off-the-shelf fc8 activation of VGGNet. Suffix '-F': the forward prediction; Suffix '-B': the backward prediction. Here we use a beam search of size 1.

	B-1	B-2	B-3	B-4	M	R	C
Bi-LSTM-F	66.2	48.5	34.3	23.6	21.5	48.7	71.0
Bi-LSTM-B	65.0	47.0	33.0	22.5	21.2	47.5	70.6
Bi-LSTM	66.3	48.7	34.9	24.5	21.9	49.1	73.9
Bi-DPN-F	67.1	49.3	34.9	24.3	21.7	49.1	71.9
Bi-DPN-B	65.9	47.8	33.6	23.1	21.3	47.8	71.8
Bi-DPN	**67.4**	**50.0**	**36.2**	**25.6**	**22.3**	**49.6**	**76.6**

Table 3: The performance of jointly fine-tuning the CNN and RNN components. Here we use a beam search of size 1.

	B-1	B-2	B-3	B-4	M	R	C
DPN	71.5	54.6	40.2	29.0	24.0	52.4	88.7
Bi-DPN	71.7	54.9	40.8	29.5	24.4	52.6	91.2

icantly improved by the fine-tuning. Notably, the fine-tuning process brings about 5 percent improvement on DPN, demonstrating the necessity of adapting visual features.

4.3. Comparison with the state-of-the-art

In Table 4, we compare our proposed scheme with state-of-the-art methods. Most of these methods are established based on VGGNet. Others are built on CNN networks which have similar or better discriminative power than VGGNet, such as GoogLeNet [25] and ResNet [11].

We can notice that: (1) Our scheme achieves better performance than most state-of-the-art methods (the upper part in Table 4). Compared with the works [8, 26, 27] which also employ the VGGNet, we obtain 1-2 percent improvement over the BLEU, ROUGE-L and CIDEr, and competitive performance over the METEOR. Notably, our approach even outperforms SCA-Res152 [8] that uses a more powerful image encoder. (2) We implement our experiments based on LRCN [4, 18]. In contrast to their latest update [18], we obtained considerably better performance than their best results. This further demonstrates the effectiveness of the DPN. (3) For benchmarking it is common practice to compare to algorithms that use similar features. Within the scope of benchmarking using the VGG feature, our algorithm had the highest performance. In the future, we would try to incorporate the more optimized features in [28, 29] to further boost the performance.

5. DISCUSSION

As we utilized three LSTM modules in DPN, it would triple the parameters for the language modelling part. Even so, we should not expect that increasing the parameters would necessarily improve the performance. To verify this, we stack three LSTM modules in the language modelling part, and utilize the target sentence to supervise each of them. We jointly train the three LSTM modules in the training phase, and average the three predictions as the final prediction in the testing phase. We name this model with deep supervision as 3LSTM-DS (we also attempt to train the model without deep supervision, but it is hard to converge).

Table 4: Comparison with the state-of-the-art. Here we use a beam search of size 4.

	B-1	B-2	B-3	B-4	M	R	C
m-RNN-VGG [6]	67.0	49.0	35.0	25.0	-	-	-
NIC [3]	-	-	-	27.7	23.7	-	85.5
LRCN [4]	66.9	48.9	34.9	24.9	-	-	-
LRCNv2 [18]	71.4	54.3	40.2	29.7	24.2	52.4	88.9
gLSTM [30]	67.0	49.1	35.8	26.4	22.7	-	81.2
Bi-LSTM [7]	67.2	49.2	35.2	24.4	-	-	-
Soft-Attention [5]	70.7	49.2	34.4	24.3	23.9	-	-
Hard-Attention [5]	71.8	50.4	35.7	25.0	23.0	-	-
RA+SF [12]	69.7	51.9	38.1	28.2	23.5	50.9	83.8
ATT-FCN [13]	70.9	53.7	40.2	30.4	24.3	-	-
ERD-VGG [31]	-	-	-	29.0	23.7	-	88.6
VAE [32]	71.0	51.0	38.0	26.0	22.0	-	89.0
Liu et al. [20]	70.7	54.8	41.0	30.4	23.8	-	89.5
Correctness [14]	-	-	37.2	27.6	24.8	-	-
SCA-VGG [8]	70.5	53.3	39.7	29.8	24.2	-	-
SCA-Res152 [8]	71.9	54.8	41.1	31.1	25.0	-	-
Ren et al. [26]	71.3	53.9	40.3	30.4	25.1	52.5	93.7
Marco et al. [27]	-	-	-	30.7	24.5	-	93.8
LSTM-A$_3$ [28]	73.5	56.6	42.9	32.4	25.5	53.9	99.8
PG-BCMR [29]	75.4	59.1	44.5	33.2	25.7	55	101.3
DPN	72.4	56.4	42.7	31.9	24.6	53.4	94.7
Bi-DPN	72.6	56.4	42.7	32.0	24.7	53.4	94.4

Table 5: Comparison between LSTM, 3LSTM-DS and DPN. Here we use a beam search of size 1.

	B-1	B-2	B-3	B-4	M	R	C
LSTM	70.0	53.0	38.6	27.4	23.6	51.5	84.3
3LSTM-DS	70.4	53.3	39	27.8	23.3	51.6	83.5
DPN	**71.5**	**54.6**	**40.2**	**29.0**	**24.0**	**52.4**	**88.7**

3LSTM-DS has the same parameter number with DPN. Following the same training and testing procedure, we compare their performance in Table 5. As can be seen, 3LSTM-DS achieves competitive results with LSTM, and inferior results than DPN. The phenomenon demonstrates that, the considerable superiority of DPN over LSTM is owing to the dual predictions, rather than increasing the parameters.

6. CONCLUSION

In this work, we propose DPN for image captioning task, which can not only improve the prediction by explicitly constraining to a low reconstruction error, but also generate an additional high-quality prediction by utilizing the reconstructed input. State-of-the-art performance is achieved by averaging both predictions. In the future, we would strive to make use of various attention models in this framework.

7. REFERENCES

[1] Vicente Ordonez, Girish Kulkarni, and Tamara L Berg, "Im2text: Describing images using 1 million captioned photographs," in *NIPS*, 2011.

[2] Polina Kuznetsova, Vicente Ordonez, Alexander C Berg, Tamara L Berg, and Yejin Choi, "Collective generation of natural image descriptions," in *ACL*, 2012.

[3] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan, "Show and tell: A neural image caption generator," in *CVPR*, 2015.

[4] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *CVPR*, 2015.

[5] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *ICML*, 2015.

[6] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan Yuille, "Deep captioning with multimodal recurrent neural networks (m-rnn)," in *ICLR*, 2015.

[7] Cheng Wang, Haojin Yang, Christian Bartz, and Christoph Meinel, "Image captioning with deep bidirectional lstms," in *ACM Multimedia*, 2016.

[8] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, and Tat-Seng Chua, "Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning," in *CVPR*, 2017.

[9] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[10] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[12] Kun Fu, Junqi Jin, Runpeng Cui, Fei Sha, and Changshui Zhang, "Aligning where to see and what to tell: Image captioning with region-based attention and scene-specific contexts," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2321–2334, 2017.

[13] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo, "Image captioning with semantic attention," in *CVPR*, 2016.

[14] Chenxi Liu, Junhua Mao, Fei Sha, and Alan L Yuille, "Attention correctness in neural image captioning.," in *AAAI*, 2017.

[15] Xinlei Chen and C Lawrence Zitnick, "Mind's eye: A recurrent visual representation for image caption generation," in *CVPR*, 2015.

[16] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele, "Grounding of textual phrases in images by reconstruction," in *ECCV*, 2016.

[17] Fangxiang Feng, Xiaojie Wang, and Ruifan Li, "Cross-modal retrieval with correspondence autoencoder," in *ACM Multimedia*, 2014.

[18] Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell, "Long-term recurrent convolutional networks for visual recognition and description," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 677–691, 2017.

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[20] Yu Liu, Yanming Guo, and Michael S Lew, "What convnets make for image captioning?," in *MMM*, 2017.

[21] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "Bleu: a method for automatic evaluation of machine translation," in *ACL*, 2002.

[22] Satanjeev Banerjee and Alon Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," in *ACL*, 2005.

[23] Chin-Yew Lin, "Rouge: A package for automatic evaluation of summaries," in *ACL*, 2004.

[24] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh, "Cider: Consensus-based image description evaluation," in *CVPR*, 2015.

[25] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015.

[26] Zhou Ren, Xiaoyu Wang, Ning Zhang, Xutao Lv, and Li-Jia Li, "Deep reinforcement learning-based image captioning with embedding reward," in *CVPR*, 2017.

[27] Marco Pedersoli, Thomas Lucas, Cordelia Schmid, and Jakob Verbeek, "Areas of attention for image captioning," in *ICCV*, 2017.

[28] Ting Yao, Yingwei Pan, Yehao Li, Zhaofan Qiu, and Tao Mei, "Boosting image captioning with attributes," in *ICCV*, 2017.

[29] Siqi Liu, Zhenhai Zhu, Ning Ye, Sergio Guadarrama, and Kevin Murphy, "Improved image captioning via policy gradient optimization of spider," in *ICCV*, 2017.

[30] Xu Jia, Efstratios Gavves, Basura Fernando, and Tinne Tuytelaars, "Guiding long-short term memory for image caption generation," in *ICCV*, 2015.

[31] Zhilin Yang, Ye Yuan, Yuexin Wu, Ruslan Salakhutdinov, and Cohen William W, "Encode, review, and decode: Reviewer module for caption generation," in *NIPS*, 2016.

[32] Yunchen Pu, Zhe Gan, Ricardo Henao, Xin Yuan, Chunyuan Li, Andrew Stevens, and Lawrence Carin, "Variational autoencoder for deep learning of images, labels and captions," in *NIPS*, 2016.