Comparison of VGG and ResNet used as Encoders for Image Captioning

Viktar Atliha
Department of Information Technologies
Vilnius Gediminas Technical University
Vilnius, Lithuania
viktar.atliha@vgtu.lt

Dmitrij Šešok
Department of Information Technologies
Vilnius Gediminas Technical University
Vilnius, Lithuania
dmitrij.sesok@vgtu.lt

***Abstract*—Recent models for image captioning are usually based on an encoder-decoder framework. Large pre-trained convolutional neural networks are often used as encoders. However, different authors use different encoder architectures for their image captioning models. This makes it more difficult to determine the effect that the encoder has on the overall model performance. In this paper we compare two popular convolution networks architectures – VGG and ResNet – as encoders for the same image captioning model in order to find out which method is the best at image representation used for caption generation.**

The results show that the ResNet outperforms VGG allowing image captioning model achieve higher BLEU-4 score. Furthermore, the results show that the ResNet allows model to achieve a score comparable with the VGG-based model with a less amount of training epochs. Based on this data we can state that encoder plays a big role and can significantly improve model without changing a decoder architecture.

***Index Terms*—image captioning, encoder-decoder framework, convolutional neural networks, VGG, ResNet**

I. Introduction

Image captioning is a very important research field on a border between computer vision and natural language processing [1]. In order to produce qualitative images descriptions image captioning system must not only understand what objects are presented, but also relationships between them. More than that, it must generate human-like sentences based on that information. Thanks to this features image captioning systems can be used in a wide range of practical tasks, such as image search, human-computer interaction and help to visually-impaired people [2], [3].

State-of-the-art models today tend to use encoder-decoder paradigm (see Fig. 1) for their architectures [4], [5], [6], [7], [8]. The goal of an encoder part is to take input image and to transform it to a vector representing that image features. The goal of a decoder part is to generate a sequence of words describing an image using that features.

Deep convolution neural networks such as VGG [9], Resnet [10] and the others pre-trained on a large image datasets are usually used as encoders. Thereby overall image captioning model will contain more implicit information about objects on an image than if it contained after training only on an image captioning dataset. This allows generate captions containing not only objects seen by overall model during training, but also seen only by encoder trained on a bigger amount of data.

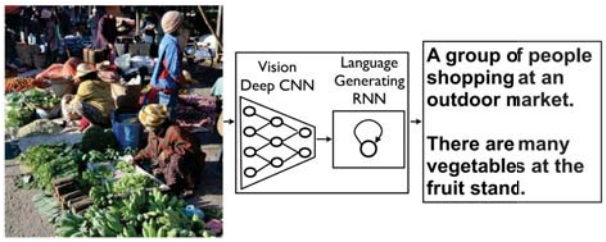

Fig. 1. Typical encoder-decoder architecture for image captioning [7].

Recurrent neural networks, such as LSTMs [11], are used as decoders in order to generate caption word by word. Using LSTM allows model save implicit information about caption part generated so far and use it during the next word generation phase. Moreover there are several approaches of caption generation. Some models firstly generate only a template with parts of speech and then generate concrete words [12]. The others generate final words straightforward [7].

More than that, image captioning task can be considered as a some kind of a ”machine translation” task that translates image to a sequence of words in some language. No wonder methods which are successful in a machine translation task are also successfully adopted to an image captioning. For example, large amount of works use attention [8], [13], [14] for models improvements, some other works use transformers [15] and so on.

One of the problem in comparison image captioning models between each other is using different encoders in different papers. Some of image captioning methods use VGG [8], [16], [17] as feature encoder, while the others use ResNet [12], [18] for that purpose or even newer encoder models [5], [19]. Authors simultaneously change both encoder and decoder models. This makes model comparison more difficult because it becomes hard to detect what influenced more: better decoder model presented by authors or just a better and newer encoder used. So it may turn out that simpler models with a better encoder can outperform more complex ones in a fare comparison.

This research was supported by Google Cloud Platform Education Programs grant

978-1-7281-9779-1/20/$31.00 ©2020 IEEE

This paper compares two image captioning models which differs only by an image encoder in order to examine exact effect caused only by encoder changing.

II. Encoders

A. VGG

VGG is an improved version of the AlexNet model. The main improvement is that filters of size 11x11 and 5x5 are replaced by a sequence of filters of smaller size 3x3. This allows to reduce the number of network parameters without changing the generalizing ability. There is a max pooling operation on a 2x2 pixel window with step 2 after some of the layers. There are three fully connected layers: the first two have 4096 channels, and the third 1000 channels after all of convolutional layers. Softmax layer is the last one. All hidden layers are followed by a ReLU activation function.

VGG has two serious drawbacks: slow learning speed and very heavy architecture in terms of memory. Nevertheless, the network is very widespread as a basic unit, allowing to obtain a high-quality representation of the image in the form of a vector.

B. ResNet

In theory an increase in the number of layers in a neural network should lead to an increase in the quality of the model as long as over-fitting is taken care of. However many architectures have the problem of "vanishing gradients". The ResNet architecture is designed to solve this problem.

It is based on the idea of shortcut connections. They make it possible to guarantee that with an increase in the number of layers of a neural network it will not need to learn the identical transformation in order to remain no worse than its counterpart with fewer layers. This is due to the fact that we immediately add a direct connection between the output of each layer with the input of the next to it layer. Thus the neural network remains to learn only residuals.

In this regard ResNet has a great advantage: it is relatively easy to optimize and thereby increase the accuracy by adding more layers.

III. Experiments

A. Dataset and Evaluation

We evaluated our method on a MS COCO dataset [20]. It contains images with 5 describing it captions for each image. Dataset consists of 82,783 training images and 40,504 validation images. We used "Karpathy" data split [6] for the performance comparison of our models. This split contains 5,000 images for validation, 5,000 images for testing and the rest are used for a training purpose. Following text processing in [8] we convert all captions to lower-case and remove words which occurs less than 5 times obtaining a captioning vocabulary of 10,010 words. As an evaluation metrics for our captioning models we have used a BLEU-4 [21] which is one of the oldest automatic metric used for image captioning evaluation. We have also used the other standard metrics for this task, including CIDEr [22], SPICE [23], METEOR [24] and ROUGE-L [25], but they showed comparable results, so we decided to report only BLEU-4 which is enough to show sufficient results. We used teacher forcing for models training, so we also report a top-5 accuracy in the next caption word generation, because it can be effectively calculated during the model training.

B. Implementation Details

Show, Attend and Tell architecture (see Fig. 2) was implemented using PyTorch.

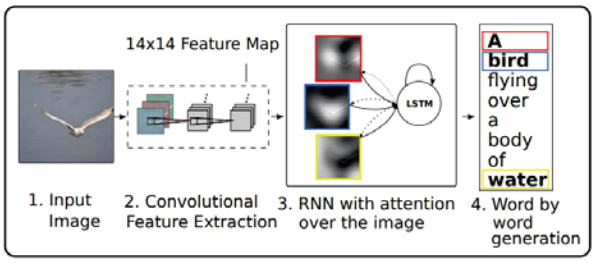

Fig. 2. Show, attend and tell architecture overview [8].

Built-in PyTorch pre-trained VGG19 and ResNet101 without fine-tuning were compared as encoders as they are the most popular architectures used as encoders for image captioning tasks. Standard LSTM with one cell was used as a decoder. Models were trained using Google Cloud Platform instance with 8 Tesla V100 GPUs, 32 core CPU and 120 GB operative memory. Input images were resized to a 256x256 pixels for uniformity. Encoder output, word embeddings and attention sizes were all set to a 512. Stochastic gradient descent with Adam optimizer with a learning rate of 0.0004 was used during training. Model were trained for 14 epochs using batch size 32 for a log-loss minimization. Teacher forcing was used during the training phase, while a beam search was used during the validation phase for captions generation. We used parameters values equal to the used in original Show, Attend and Tell paper.

C. Performance Comparison and Analysis

During the training both models performed with a very similar results both in accuracy and in log-loss, although the model with ResNet encoder showed a slightly better results. The model with VGG encoder has reached 0.76516 accuracy, while for the model with ResNet encoder has reached 0.77139 (see Fig. 3). Models have reached log-losses 3.1241 and 3.0838 respectively.

However, models showed more different results during the validation phase. Model with ResNet encoder showed better results not only on a directly minimized log-loss, but also on an accuracy (see Fig. 4) and, what is more important, on an image caption specific metrics BLEU-4 (see Fig. 5). The difference in BLEU-4 is 0.008 which is rather significant. More than that, model with ResNet as encoder showed better speed of training than model with VGG as encoder. The first

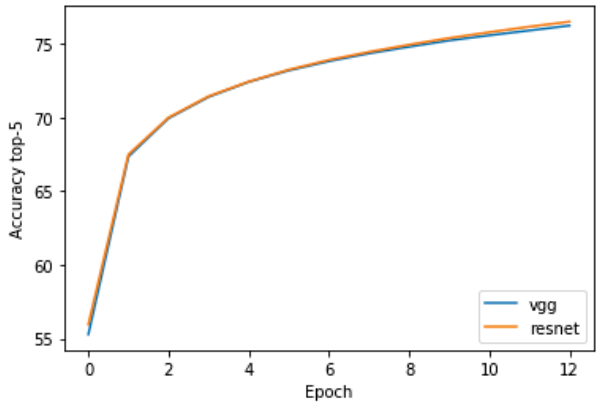

Fig. 3. Train accuracy top-5.

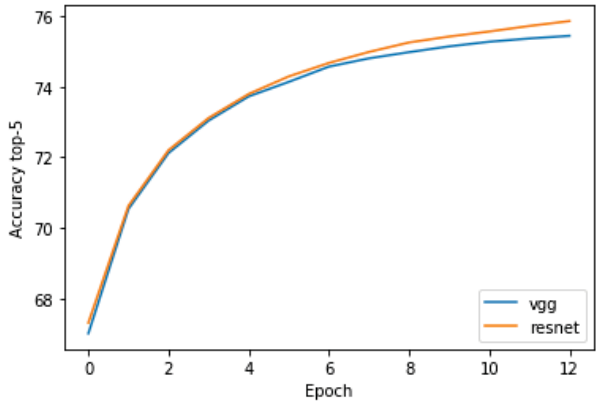

Fig. 4. Validation accuracy top-5.

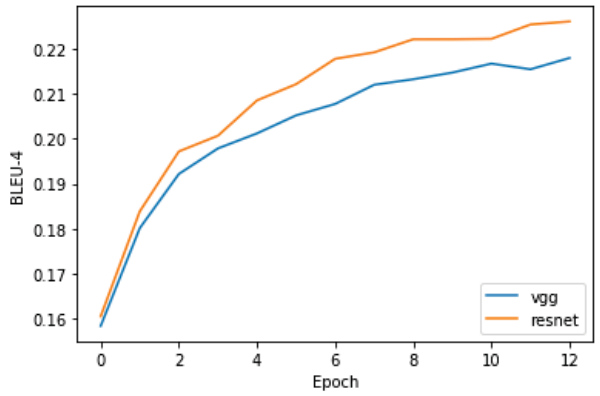

Fig. 5. Validation BLEU-4.

model reached 0.21 in BLEU-4 on a 5 epoch, while the second reached it only on a 7 epoch.

Summarized results both for training and validation are given in a Table I.

Based on the results presented above, it can be concluded that ResNet performed better than VGG as an encoder for the image captioning task. This may be due to the fact that ResNet is known as a stronger architecture for solving many problems associated with image processing. Skip connections in it architecture allow model to learn more efficiently and its greater depth compared to VGG allows it to capture more complex concepts which makes it a more efficient encoder.

TABLE I
MODELS COMPARISON

Phase	Encoder	Top-5 accuracy	Log-loss	BLEU-4
Training	VGG	0.762	3.146	-
Training	ResNet	**0.765**	**3.129**	-
Validation	VGG	0.754	3.216	0.218
Validation	ResNet	**0.759**	**3.189**	**0.226**

IV. CONCLUSION

In this paper we compared two different image encoders such as VGG and ResNet for an image captioning task by analyzing BLEU-4 score at validation phase and top-5 accuracy and log-loss at both training and validation phases. We analyzed both an absolute value of mentioned metrics and it dependency based on a number of training epoch. As a result, model with a ResNet encoder outperformed model with a VGG encoder by 0.008 in BLEU-4 score and reached 0.21 score two epochs faster.

REFERENCES

[1] R. Staniūtė and D. Šešok, "A systematic literature review on image captioning," *Applied Sciences*, vol. 9, no. 10, p. 2024, 2019.
[2] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2425–2433.
[3] A. Das, S. Kottur, K. Gupta, A. Singh, D. Yadav, J. M. Moura, D. Parikh, and D. Batra, "Visual dialog," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 326–335.
[4] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7008–7024.
[5] K. Shuster, S. Humeau, H. Hu, A. Bordes, and J. Weston, "Engaging image captioning via personality," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 516–12 526.
[6] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3128–3137.
[7] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3156–3164.
[8] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.
[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[12] J. Lu, J. Yang, D. Batra, and D. Parikh, "Neural baby talk," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7219–7228.
[13] L. Huang, W. Wang, Y. Xia, and J. Chen, "Adaptively aligned image captioning via adaptive attention time," in *Advances in Neural Information Processing Systems*, 2019, pp. 8940–8949.

[14] W. Wang, Z. Chen, and H. Hu, "Hierarchical attention network for image captioning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 8957–8964.

[15] G. Li, L. Zhu, P. Liu, and Y. Yang, "Entangled transformer for image captioning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 8928–8937.

[16] Y. Li, T. Yao, Y. Pan, H. Chao, and T. Mei, "Pointing novel objects in image captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 497–12 506.

[17] G. Yin, L. Sheng, B. Liu, N. Yu, X. Wang, and J. Shao, "Context and attribute grounded dense captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6241–6250.

[18] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6077–6086.

[19] X. Zhu, L. Li, J. Liu, H. Peng, and X. Niu, "Captioning transformer with stacked attention modules," *Applied Sciences*, vol. 8, no. 5, p. 739, 2018.

[20] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[21] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[22] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4566–4575.

[23] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Spice: Semantic propositional image caption evaluation," in *European Conference on Computer Vision*. Springer, 2016, pp. 382–398.

[24] S. Banerjee and A. Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," in *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005, pp. 65–72.

[25] C.-Y. Lin and F. J. Och, "Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics," in *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2004, p. 605.