Boosted Attention: Leveraging Human Attention for Image Captioning (Supplementary Material)

Shi Chen and Qi Zhao

Department of Computer Science and Engineering,
University of Minnesota
{chen4595,qzhao}@umn.edu

This supplementary material provides detailed procedure for experiments discussed in main paper and additional analysis on defining hyper-parameter. More specifically, we elaborate 1) the process for generating captioning attention maps, and 2) the reasons behind the selection of hyper-parameter ϵ in the proposed Boosted Attention method.

1 Generating Captioning Attention Map

In this section, we discuss our experimental procedure on generating the captioning attention map. Similar as [1], our captioning attention maps are generated based on visual object category to sentence's noun (VOS) mapping. More specifically, we firstly identify all nouns within the captions using Stanford POS tagger [2] and then compute the similarity between these nouns and the unique categories in MSCOCO [3] based on word2vec [4]. Nouns with similarity scores above a threshold (0.18, the same as [1]) are considered valid and matched with the categories. For each sentence, we utilize the object masks of MSCOCO categories to construct the attention map for image captioning. Since the attention for image captioning is affected by the referral order of nouns and the size of object corresponding to nouns as mentioned in [1], after obtaining the attention maps by projecting the object masks corresponding to categories on the images, we further normalize the masks for different categories using a weight w computed based on the referral order of corresponding nouns and size of the masks:

$$w_i = \frac{S_i}{S} \cdot \frac{n}{i} \tag{1}$$

where S_i and S denote the size of the mask for the i_{th} noun and the size of the image, n represents the total number of nouns within the caption. Since MSCOCO provides 5 captions for every image, we accumulate the attention maps for all captions corresponding to an image and normalize the map to values between 0 and 1. After computing the attention maps for image captioning, we compare them with ground-truth saliency maps from the SALICON dataset [5]. Totally 9889 images from the training set of SALICON are evaluated, the other 111 maps are invalid because all of the nouns within captions fail to be matched with MSCOCO categories.

2 Parameter Selection for Boosted Attention Method

In the main paper, we propose the Boosted Attention method that utilizes an asymmetric function to integrate the stimulus-based features with the original visual features from ResNet-101 [6] as follows:

$$I^{'} = W_v I \circ log(W_{sal} I + \epsilon) \tag{2}$$

in which $\circ$ represents hadamard product, W_v is a convolutional layer that further encodes visual features, W_{sal} denotes the stimulus-based attention and ϵ is a hyper-parameter. The function constructs a residual module where ϵ determines how much of the original visual features we would like to preserve and W_{sal} controls the residual term. While in the paper we use $\epsilon = e$ to preserve the identity of visual features, in this section we provide more details on the selection of ϵ.

We evaluate and demonstrate the effects of different hyper-parameter choices by training a set of models with corresponding hyper-parameters. The training procedure is the same as in the experiment section of main paper, except that for efficiency we do not apply reinforcement learning and only compare the models trained under supervised learning. Six typical options with different properties are selected for ϵ and compared in this experiment:

- $\boldsymbol{\epsilon = 0}$: solely rely on the stimulus-based attention to control the contributions of visual features and allow the visual features to be negative. Note that to ensure the term inside logarithm to be positive, we set $\epsilon = 10^{-3}$ instead of exact 0.
- $\boldsymbol{\epsilon = 1}$: solely rely on the stimulus-based attention to control the contributions of visual features but ensure that the visual features are non-negative.
- $\boldsymbol{\epsilon = e^{\frac{1}{2}}}$: construct a residual module and preserve part of visual features.
- $\boldsymbol{\epsilon = e}$: construct a residual module and preserve the identity of original visual features.
- $\boldsymbol{\epsilon = e^{2}}$: construct a residual module and reduce the contributions of stimulus-based attention.
- $\boldsymbol{\epsilon = e^{4}}$: construct a residual module and more significantly reduce the contributions of stimulus-based attention.

According to the results on Karpathy's test split [7] (Figure 1), using $\epsilon = e$ for preserving the identity of visual features provides the best performance among all of the evaluation metrics. Without intentionally keeping the original visual features, *i.e.* setting $\epsilon = 0$ or 1, the performance is degraded significantly especially on the CIDEr score, showing the importance of preserving the original visual features. Since stimulus-based attention may not attend to all of the regions of interest relevant to generating current caption, instead of discarding ($\epsilon = 1$) or suppressing ($\epsilon = 0$) the visual features that are not fixated by the attention, it is necessary to keep them for sub-sequential computation and let both types of attentions (*i.e.* stimulus-based attention and top-down attention) determine their contributions. Moreover, by preserving portion of original visual

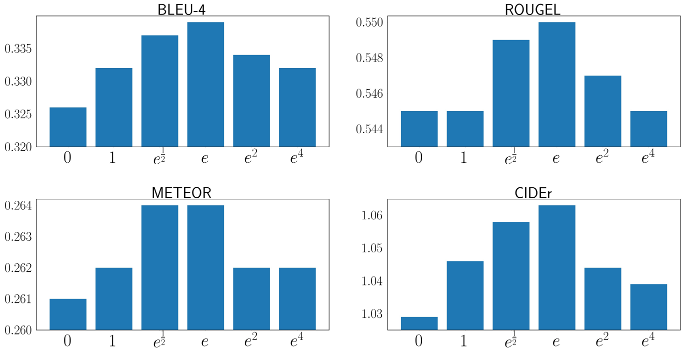

Fig. 1. Evaluation scores for models with different selections of hyper-parameter ϵ. The x-axis denotes the values of ϵ while y-axis represents the corresponding scores in terms of each metric.

features with $\epsilon = e^{\frac{1}{2}}$, the performance is significantly improved and only slight performance loss is observed compared to preserving the identity of features, further indicating the effectiveness of preserving original visual features.

Besides showing the importance of preserving original visual features (corresponding to results for $\epsilon = 0, 1, e^{\frac{1}{2}}, e$), the results also indicate the influences of stimulus-based attention on the overall performance of the model. More specifically, by utilizing relatively large ϵ values ($\epsilon = e^2, e^4$) to limit the variance of the log term in Equation 2, we intentionally reduce the effects of stimulus-based attention. In Figure 1, we can clearly see that the model performance decreases monotonically with the increase of ϵ (from $\epsilon = e$ to $\epsilon = e^4$), which demonstrates that the influences of stimulus-based attention are essential to achieve satisfying performance.

Based on the aforementioned observations, we define the hyper-parameter ϵ as a mathematical constant e in the proposed Boosted Attention method, which balances the contributions between the original visual features and stimulus-based attention.

References

1. Tavakoliy, H.R., Shetty, R., Borji, A., Laaksonen, J.: Paying attention to descriptions generated by image captioning models. In: 2017 IEEE International Conference on Computer Vision (ICCV). (2017) 2506–2515
2. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J., McClosky, D.: The Stanford CoreNLP natural language processing toolkit. In: Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations. (2014) 55–60
3. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollr, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European Conference on Computer Vision (ECCV), Z*ü*rich (2014) Oral.
4. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR **abs/1301.3781** (2013)
5. Jiang, M., Huang, S., Duan, J., Zhao, Q.: Salicon: Saliency in context. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2015) 1072–1080
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. CoRR **abs/1512.03385** (2015)
7. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2015) 3128–3137